

# Installation, Betrieb und Wartung imperia 9

Imperia AG  
Von-der-Wettern-Straße 27  
D - 51149 Köln  
<http://www.imperia.net>

---





Installation, Betrieb und Wartung  
**imperia 9**

Auflage: 11.01.2012 17:41 [1420]

Copyright © 2001-2012 Imperia AG Hürth/Germany

Alle Rechte vorbehalten. Dieses Handbuch darf ohne vorherige schriftliche Genehmigung der Imperia AG weder vollständig noch auszugsweise kopiert, fotokopiert, vervielfältigt, übersetzt oder in eine elektronische oder maschinenlesbare Form übertragen werden.

|  |    |
|--|----|
| 1. Einführung .....  | 1  |
| 1.1. Konventionen in dieser Dokumentation .....                            | 1  |
| 2. Installation von <b>imperia</b> <sup>i9</sup> ✓ .....                   | 2  |
| 2.1. Nicht länger unterstützte Module <sup>i9</sup> + .....                | 3  |
| 2.2. Voraussetzungen .....   | 3  |
| 2.2.1. Dateisystem .....   | 3  |
| 2.2.2. Browser .....   | 4  |
| 2.2.2.1. Browsermodule <sup>i9</sup> + .....                               | 4  |
| 2.2.3. Perl .....  | 5  |
| 2.2.4. Perl-Modul BerkeleyDB <sup>i9</sup> ✓ .....                         | 5  |
| 2.2.5. Perl-Modul DB_File .....  | 6  |
| 2.2.6. Perl-Modul DBI .....  | 6  |
| 2.2.7. Perl-Modul DBD::SQLite .....  | 6  |
| 2.2.8. HTML::Parser .....  | 6  |
| 2.2.9. HTML/FormatText/WithLinks.pmr .....                                 | 6  |
| 2.2.10. XML::Dom .....   | 7  |
| 2.2.11. Time::HiRes .....  | 7  |
| 2.2.12. HTML::FormatText::WithLinks .....                                  | 7  |
| 2.2.13. ImageMagick .....  | 7  |
| 2.2.14. Webserver .....  | 7  |
| 2.2.15. Zusätzliche Voraussetzungen für SQL-basierte Datenhaltung .....    | 8  |
| 2.2.15.1. Perl-Module .....  | 8  |
| 2.2.15.2. Oracle® .....  | 8  |
| 2.2.15.3. MS-SQL® .....  | 8  |
| 2.2.15.4. MySQL .....  | 8  |
| 2.2.15.5. PostgreSQL .....   | 9  |
| 2.2.15.6. SQLite .....   | 9  |
| 2.2.16. Weitere Informationen .....  | 9  |
| 2.3. Installation unter UNIX®-Derivaten .....                              | 9  |
| 2.4. Installation unter Windows® 2000 / 2008 Server / XP / Vista / 7 ..... | 10 |
| 2.5. Installation eines Zielsystems .....                                  | 11 |
| 2.6. Installieren eines Patches ("Hotfix") .....                           | 11 |
| 2.7. Update von einer älteren <b>imperia 9</b> -Version .....              | 12 |
| 2.7.1. Updaten der Datenbankstruktur .....                                 | 12 |
| 2.8. Konfiguration von <b>imperia</b> .....                                | 12 |
| 2.8.1. ALLOW_TEMPLATE_FORM .....   | 12 |
| 2.8.2. ALLOW_TEMPLATE_SCRIPT .....   | 12 |
| 2.8.3. ALLOW_TEMPLATE_FRAME .....  | 12 |
| 2.8.4. UPLOAD_OBJECTS .....  | 12 |
| 2.8.5. OPERATING-SYSTEM .....  | 13 |
| 2.8.6. CGI-DIR .....   | 13 |
| 2.8.7. DOCUMENT-ROOT .....   | 13 |
| 2.8.8. CGI-BIN .....   | 13 |
| 2.8.9. DEMOMODE .....  | 13 |
| 2.8.10. SERVER-SCHEME .....  | 13 |
| 2.8.11. SERVER_NAME .....  | 13 |
| 2.8.12. SERVER-PORT .....  | 14 |
| 2.8.13. HTTP_HOST .....  | 14 |
| 2.8.14. LOCK_EX, LOCK_SH, LOCK_UN, LOCK_NB .....                           | 14 |
| 2.8.15. COOKIE_TTL .....   | 14 |
| 2.8.16. SAFE_TEMPLATE .....  | 14 |
| 2.8.17. REG_KEY, REG_KEY2, REG_NAME .....                                  | 14 |
| 2.8.18. IMPERIA_ERROR_LOG .....  | 14 |
| 2.8.19. LATCHING .....   | 14 |
| 2.9. Modul-Installation .....  | 14 |
| 2.9.1. Verwendung des Grids .....  | 14 |
| 2.9.2. OneClickEdit installieren .....                                     | 15 |
| 2.10. Smart Database Update .....  | 15 |
| 2.10.1. Update und Wartung .....   | 15 |
| 2.10.1.1. site_doctree_admin.pl .....                                      | 15 |
| 2.10.1.2. site_doctree_repair.pl .....                                     | 16 |

|  |    |
|--|----|
| 2.10.2. Spachunterstützung im Betriebssystem  | 16 |
| 3. Migration                                  | 18 |
| 3.1. Vorbereitung  | 18 |
| 3.2. Migrationsschritte  | 18 |
| 3.2.1. Datenhaltung File65   | 19 |
| 3.2.2. Datenhaltung SQL-DB   | 20 |
| 3.3. Nachbereitung   | 21 |
| 4. Betriebskonzept                            | 22 |
| 4.1. Ausführungsschicht  | 23 |
| 4.1.1. Apache mod_perl   | 23 |
| 4.1.2. Fast-CGI  | 24 |
| 4.1.3. Interner Webserver                     | 24 |
| 4.2. Speicherort   | 25 |
| 4.2.1. Lokal   | 25 |
| 4.2.2. SAN   | 25 |
| 4.2.3. NAS   | 25 |
| 4.3. Datenbanken   | 25 |
| 4.3.1. PostgreSQL  | 25 |
| 4.3.2. Oracle  | 26 |
| 4.3.3. MySQL   | 26 |
| 4.3.4. MS-SQL  | 26 |
| 4.4. Paradigmen  | 26 |
| 4.4.1. Entwicklungssystem  | 26 |
| 4.4.2. Livesystem  | 27 |
| 4.5. Standardsysteme   | 27 |
| 4.5.1. Minimal   | 27 |
| 4.5.2. Minimal Standby   | 28 |
| 4.5.3. Standard Filesystem   | 28 |
| 4.5.4. SQL-Datenhaltung  | 29 |
| 4.5.5. Loadbalancer, SQL-Datenhaltung  | 31 |
| 4.5.6. Loadbalancer für Basis- und Livesystem  | 31 |
| 4.6. Systeme mit Teasermanagement  | 31 |
| 4.6.1. Variante 1  | 32 |
| 4.6.2. Variante 2  | 33 |
| 4.6.3. Variante 3  | 34 |
| 4.7. Testsysteme   | 34 |
| 4.8. Performance   | 35 |
| 4.8.1. Hardware  | 35 |
| 4.8.2. Betriebssystem  | 35 |
| 4.8.3. Filesystem  | 35 |
| 4.8.4. Datenhaltung  | 35 |
| 4.8.5. Browser   | 35 |
| 4.8.6. Datenstruktur   | 35 |
| 5. Wartung   | 36 |
| 5.1. Backup  | 36 |
| 5.2. Log-Dateien   | 36 |
| Index  | 37 |

---

# Tabellenverzeichnis

|   |    |
|---|----|
| 3.1. ....   | 19 |
| 3.2. ....   | 19 |
| 3.3. ....   | 19 |
| 3.4. ....   | 19 |
| 3.5. ....   | 20 |
| 3.6. ....   | 20 |
| 3.7. ....   | 20 |
| 3.8. ....   | 20 |
| 3.9. ....   | 21 |
| 3.10. ....  | 21 |
| 4.1. Konfiguration „Minimal-System“ .....                                 | 28 |
| 4.2. Konfiguration „Minimal Standbysystem“ .....                          | 28 |
| 4.3. Konfiguration „Standard Filesystem“ .....                            | 29 |
| 4.4. Konfiguration Webserver „SQL-Datenhaltung“ .....                     | 29 |
| 4.5. Konfiguration Datenbankserver „SQL-Datenhaltung“ .....               | 30 |
| 4.6. Konfiguration Webserver „Loadbalancer, SQL-Datenhaltung“ .....       | 30 |
| 4.7. Konfiguration Datenbankserver „Loadbalancer, SQL-Datenhaltung“ ..... | 30 |

# Kapitel 1. Einführung

Dieses Handbuch gibt einen Überblick über die Vorgehensweise bei der Installation von **imperia 9**. Dazu zählen die Überlegungen bei der Erstinstallation, die Betriebskonzepte, sowie die tatsächliche Installation bzw. Migration und die Wartung des Systems.

## 1.1 Konventionen in dieser Dokumentation

Die vorliegende Dokumentation benutzt folgende Konventionen:

- Dialoge, Abschnitte, Eingabefelder, Markierungsfelder, kurz: alle Programmbestandteile werden `entsprechend formatiert` dargestellt.
- Auf besondere Zusammenhänge wird mit der Überschrift „**Achtung:**“ oder „**Warnung:**“ hingewiesen.
- Hinweise werden **fett** dargestellt.
- Code-Beispiele und Scriptaufrufe werden in `Festbreitenschrift` dargestellt.

## Kapitel 2. Installation von imperia

Die Installation von **imperia** kann unter folgenden Betriebssystemen erfolgen:

- UNIX® / UNIX® -Derivate (SuSE, RedHat, ...)
- Windows® 2000 / 2003 / XP

Die Installation wird mit Hilfe des interaktiven Installations-Skripts `install.pl` durchgeführt, mit dem sowohl das Basissystem als auch die Zielsysteme installiert werden können.



### Hinweis:

*Die wichtigsten Änderungen bei der Installation von **imperia 9** gegenüber den Vorgängerversionen sind, dass der Installer das bestehende System (dev/live) selbst erkennt.*

*Bei der Verwendung von `mod_perl` wird das CGI-Verzeichnis nicht benötigt, so dass es bei der Installation nicht mit angegeben wird. In diesem Fall sollte das site-Verzeichnis angegeben werden.*

Das Skript zur Installation kann mit folgenden Optionen gestartet werden:

### **-h | -help**

Zeigt die Liste der Optionen.

### **-l | -live \***

Installiert ein Zielsystem. Lesen Sie hierzu auch Abschnitt 2.5 **Installation eines Zielsystems** auf Seite 11.



### Hinweis:

*Das CGI-BIN Verzeichnis wird unter `mod_perl` nicht mehr benötigt! Geben Sie statt des das site Verzeichnis mit an.*

### **-c | -cgibin Pfad \***

Absoluter Pfad zum CGI-BIN-Verzeichnis. In `mod_perl` Umgebungen kann für diesen Wert „none“ angegeben werden, da unter `mod_perl` `cgi` nicht benötigt wird, Sie können jedoch `mod_perl` und `cgi` parallel betreiben, wenn Sie z.B. eigene Skripte haben.

### **-s | -site Pfad \***

Absoluter Pfad zum SITE-Verzeichnis.

### **-d | -docroot Pfad \***

Absoluter Pfad zum Document-Root-Verzeichnis.

### **-w Location Pfad \***

URL der CGI-BIN-Location des Webservers.

### **-k | -keep**

Überschreibt nie bereits vorhandene Dateien, sondern legt Backup-Kopien dieser Dateien an.

### **-r | -rm | -remove**

Überschreibt immer bereits vorhandene Dateien und legt nie Backup-Kopien dieser Dateien an.

## **-f | -force**

Erzwingt Neu-Installation der Konfigurations-Verzeichnisse und Dateien.

## **-n | -noprompt**

Installation ohne Bestätigung der einzelnen Punkte durch den User.

## **-p**

Bewahrt die Dateien `system.conf` und `imperia.conf` auch dann, wenn die dort gefundenen Pfade von denen über die Kommandozeile oder User-Eingabe übergebenen Pfaden abweichen.

## **-rf | -fr**

Gleichzeitig `-force` und `-remove`.



### **Hinweis:**

*Optionen, die mit einem Stern gekennzeichnet sind, brauchen Sie nur zu verwenden, wenn Sie die Abfrage der entsprechenden Option während des Installationsprozesses überspringen wollen.*

Alle vom Installationskript ausgeführten Aktionen werden in einem Installationsprotokoll (`install.log`) festgehalten, das in dem Verzeichnis abgelegt wird, in dem das Installations-Skript gestartet wurde.

In diesem Protokoll werden alle Aktionen des Skripts und Ihre Eingaben gespeichert. Halten Sie diese Datei für eventuelle Supportanfragen bereit.

## **2.1 Nicht länger unterstützte Module** <sup>19</sup>

In **imperia 9** werden die Partner-Plug-Ins „Resubmit“, „Escalation“ und „Mail“ von der Firma Netpioneer nicht weiter unterstützt. Die Imperia AG bietet nun eigene Module mit dieser Funktionalität an.

Das EWE (Easy Word Modul) der Firma Interface Projects wird nicht mehr in **imperia** unterstützt. Für das EWE wird der Kompatibilitäts-Modus intern aktiviert, so dass alle Templateaufrufe vom iWE behandelt werden. Die System\_Conf EWE\_COMPAT\_MODE wird entfernt und hat keine Funktion mehr.

## **2.2 Voraussetzungen**

Um **imperia** installieren zu können, müssen für alle Betriebssysteme folgende Voraussetzungen geschaffen werden:

### **2.2.1 Dateisystem**

Die Performance von **imperia** hängt — insbesondere bei der Verwendung der Datenhaltung File65 — stark vom verwendeten Dateisystem ab. Folgende Faktoren sollten berücksichtigt werden:

- Das Dateisystem sollte für die Verwaltung vieler, kleiner Dateien optimiert sein. Unter Linux muss daher von der Verwendung von ext3 (oder ext2) dringend abgeraten werden. Bewährt haben sich die Dateisysteme Ext4, ReiserFS und XFS.
- Die Verwendung von Netzwerkdateisystemen, insbesondere NFS, wirkt sich negativ auf die Performance aus. Sollte NFS zum Einsatz kommen, sind Tuningmaßnahmen nötig.
- Der in `site/config/system.conf` konfigurierte Pfad für das Site-Verzeichnis sollte kein symbolischer Link sein.
- Es empfiehlt sich weitere dateisystemspezifische Optimierungsmaßnahmen, zum Beispiel Defragmentierung, regelmäßig durchgeführt werden.
- Die Leistungsfähigkeit aller Dateisysteme leidet unter Betrieb an der Kapazitätsgrenze. Eine Belegung von mehr als 90 Prozent des verfügbaren Platzes sollte dringend vermieden werden.

## 2.2.2 Browser

In **imperia 9** werden die folgenden Browser wie folgt unterstützt.

### Aktuelle Browser

Aktuelle Browser werden vollständig unterstützt, da diese erst die Möglichkeit zu aktuellen Technologien bieten.

MS Internet Explorer 7 und 8

Mozilla Firefox 3X auf Windows, Linux und Mac OS X

### Aktuelle Browser mit geringer Verbreitung

Aktuelle Browser mit geringer Verbreitung werden zunächst nicht komplett unterstützt. Grundsätzlich kann **imperia** mit diesen Browsern bedient werden, wobei teilweise Darstellungs- und Funktionsprobleme aufgrund abweichender Implementierungen auftreten können.

Safari auf Mac OS X und Windows

Google Chrome auf Mac OS X und Windows

Opera

Konquerer

### Alte Browser

Alte Browser ( $\geq 10$  Jahre) werden nur noch teilweise unterstützt. Die meisten Grundfunktionen in **imperia** funktionieren mit diesen Browsern. Da diese Browser nicht die Voraussetzungen für moderne Web-Applikationen mitbringen, ist nicht gewährleistet, dass alle Bestandteile der Software zuverlässig und performant funktionieren.

MS Internet Explorer 6

### Sehr alte Browser

Sehr alte Browser werden nicht unterstützt. Diese Browser verfügen nicht über die Voraussetzungen für moderne Web-Applikationen.

MS Internet Explorer 5

Netscape 4.7

### 2.2.2.1 Browsermodule <sup>19</sup>

#### JavaFX für das Grid

Der Rechner auf dem der Browser läuft, muss bei der ersten Verwendung des Grids eine Internetverbindung haben, um das JavaFX Plugin zu installieren.

#### Flash für den Mehrfachupload

Sollen Assets über den Mehrfachupload in das MAM geladen werden können, so muss auf dem Rechner auf dem der Browser das Flash Browser-Plugin installiert werden.



#### Hinweis:

*Stellt der Browser (besonders IE) fest, dass die in der Seite enthaltenen Elemente nicht dem Doctype entsprechen, oder der Doctype ganz fehlt, so geht der Browser in den sogenannten „Quirks“-Modus. Dabei werden die Kontroll-Elemente der Seite ggf. falsch angezeigt.*

*Wenn die Seite kein valides HTML bzw. XHTML beinhaltet bzw. diese Inhalte nicht zum Doctype passen, dann schaltet der Browser ebenfalls in den „Quirks“-Modus. Sind nur wenige Abweichungen enthalten, stellen die meisten Browser den Inhalt jedoch noch richtig dar.*

Geschieht dies im Template, so werden die **imperia 9** Kontroll-Elemente ggf. nicht richtig dargestellt.

Abhilfe:

a) Der Templateentwickler muss den „Quirks“-Modus vermeiden => Wenn die Seite einen Doctype angibt und die Seite valides HTML bzw. XHTML beinhaltet und zum Doctype passt, tritt dieses Problem nicht auf.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

b) Möchte oder kann der Templateentwickler dies aus bestimmten Gründen nicht, so kann er einen View mit einer eigenen einfachen Darstellung anlegen, in dem der „Quirks“-Modus keine Auswirkung hat.

### 2.2.3 Perl

Installieren Sie die Perl-Version 5.8.1., 5.10 oder eine höhere Version.



**Achtung:**

**imperia** benötigt zwingend eine Perl-Version ab Perl 5.8.

Für Windows® -Betriebssysteme können Sie Perl unter folgenden Adressen herunterladen:

<http://www.activestate.com/>



**Wichtig:**

Verwenden Sie bitte unbedingt die Windows-Installer-Version des ActiveState Interpreters (\*.msi), da bei der Batch-Installation Probleme mit der Ausführung von CGI-Skripts auftreten sind.

### 2.2.4 Perl-Modul BerkeleyDB <sup>19</sup> ✓

Das Perl-Modul BerkeleyDB brauchen Sie für den Betrieb der **imperia** Volltextsuche. Sie können es unter folgender URL kostenlos herunterladen:

<http://www.perl.com/CPAN/modules/by-module/BerkeleyDB/>

Für Active-State-Perl unter Windows bietet sich die Installation mittels ppm an.

Die BerkeleyDB ist ein Modul, das für die **imperia** Volltextsuche benötigt wird. Früher war dieses Modul direkt im Internet frei verfügbar. Nachdem Oracle die Herstellerfirma gekauft hat, kann die BerkeleyDB nur noch über Oracle bezogen werden. Verwenden Sie bitte die Datenbankversion 4.5.20, welche unter folgendem Link heruntergeladen werden kann:

<http://download.oracle.com/berkeley-db/db-4.5.20.msi>

Der Start des Downloads dauert einen moment, aber er startet.

Sollte trotz erfolgreicher Installation die DLL `libdb45.dll` nicht gefunden werden, so müssen Sie den Pfad zur BerkeleyDB Installation in Ihrem Systempfad hinzufügen.

## 2.2.5 Perl-Modul DB\_File

Für den Betrieb von **imperia** benötigen Sie das Perl-Modul DB\_File, das Sie vom CPAN herunterladen und installieren können.

[http://www.perl.com/CPAN/modules/by-module/DB\\_File/](http://www.perl.com/CPAN/modules/by-module/DB_File/)

Für Active-State-Perl unter Windows bietet sich die Installation mittels ppm an.

## 2.2.6 Perl-Modul DBI

Damit Sie das Modul DBD::SQLite nutzen können, benötigen Sie das modul DBI. Mindestversion ist 1.21. Sie können das Modul unter folgender URL herunterladen:

<http://search.cpan.org/~timb/DBI/DBI.pm>

Für Active-State-Perl unter Windows bietet sich die Installation mittels ppm an.

Öffnen Sie eine DOS-Eingabeaufforderung und geben Sie folgenden Befehl ein:

```
ppm install DBI
```

## 2.2.7 Perl-Modul DBD::SQLite

Das Perl-Modul DBD::SQLite brauchen Sie für die dateisystembasierte Datenhaltung File65. Sie können es unter folgender URL kostenlos herunterladen. Bitte verwenden Sie immer die aktuellste Version >= 1.29 (DBD-SQLite-Version.tar.gz).

<http://search.cpan.org/CPAN/authors/id/M/MS/MSERGEANT/>

Für Active-State-Perl unter Windows bietet sich die Installation mittels ppm an.



### **Wichtig:**

*Zum Betrieb dieses Moduls muss das Modul DBI mindestens in Version 1.21 vorliegen.*

## 2.2.8 HTML::Parser

HTML::Parser (C-Version)

Für Active-State-Perl unter Windows bietet sich die Installation mittels ppm an.

Öffnen Sie eine DOS-Eingabeaufforderung und geben Sie folgenden Befehl ein:

```
ppm install HTML-Parser
```

## 2.2.9 HTML/FormatText/WithLinks.pmr

HTML/FormatText/WithLinks.pm

Für Active-State-Perl unter Windows bietet sich die Installation mittels ppm an.

Öffnen Sie eine DOS-Eingabeaufforderung und geben Sie folgenden Befehl ein:

```
ppm install HTML-FormatText-WithLinks
```

### 2.2.10 XML::Dom

XML::Dom (mit allen benötigten Abhängigkeiten)

Für Active-State-Perl unter Windows bietet sich die Installation mittels ppm an.

Öffnen Sie eine DOS-Eingabeaufforderung und geben Sie folgenden Befehl ein:

```
ppm install XML-Dom
```

### 2.2.11 Time::HiRes

Time::HiRes (soweit noch nicht in der Perl-Distribution enthalten)

Dieses Modul sollte, wenn es nicht schon mit Perl installiert wurde, einfach zu installieren sein.

Für Active-State-Perl unter Windows bietet sich die Installation mittels ppm an.

Öffnen Sie eine DOS-Eingabeaufforderung und geben Sie folgenden Befehl ein:

```
ppm install Time-HiRes
```

### 2.2.12 HTML::FormatText::WithLinks

Dieses Modul sollte, wenn es nicht schon mit Perl installiert wurde, einfach zu installieren sein.

Für Active-State-Perl unter Windows bietet sich die Installation mittels ppm an.

Öffnen Sie eine DOS-Eingabeaufforderung und geben Sie folgenden Befehl ein:

```
ppm install HTML-FormatText-WithLinks
```

### 2.2.13 ImageMagick

Für die Verwendung des Media Asset Managements (MAM) benötigen Sie **ImageMagick**. Unter Linux Systemen sollten Sie den Paketmanager zur Installation von ImageMagick verwenden. Ansonsten können Sie es unter folgender Adresse herunterladen:

<http://www.imagemagick.com/www/binary-releases.html>

Bei vielen Linux- und UNIX®-Distributionen ist ImageMagick Bestandteil der Distribution und sollte wie jedes andere Programm über die entsprechende Paketverwaltung installiert werden.



#### **Achtung:**

*Bei der "Enterprise Edition" der SuSE AG ist ImageMagick nicht Bestandteil der Distribution, sondern muss nachinstalliert werden. Bei der Professional Edition fehlt das Modul PerlMagick — die Schnittstelle zu Perl.*

Neben der Installation von ImageMagick muss auch das Perl-Modul für ImageMagick installiert werden.



#### **Hinweis:**

*Bitte beachten Sie, dass unter Windows der Pfad zur ImageMagick in den Systempfad eingetragen werden muss.*

### 2.2.14 Webserver

Installieren Sie einen Webserver. Wir empfehlen den Apache Webserver, da dieser unter CGI, FAST-CGI oder mod\_perl betrieben werden kann.

<http://httpd.apache.org>



### Wichtig:

*Bitte beachten Sie, dass Domainnamen nach RFC keine Unterstriche enthalten dürfen. Der Internet Explorer akzeptiert in diesem Fall keine Cookies. **imperia** benötigt jedoch Cookies für den Betrieb.*

## 2.2.15 Zusätzliche Voraussetzungen für SQL-basierte Datenhaltung

Wenn Sie ein SQL-RDBMS für die Datenhaltung einsetzen möchten, müssen zusätzlich folgende Voraussetzungen erfüllt sein:



### Hinweis:

*Bitte beachten Sie, dass der Webserver neu gestartet (mindestens aber neu geladen) werden muss, wenn die Verbindung zur Datenbank (insbesondere MySQL) unterbrochen oder die Datenbank neu gestartet wurde.*

### 2.2.15.1 Perl-Module

- DBD::\* - Das passende Modul für das eingesetzte RDBMS, Download über [www.cpan.org](http://www.cpan.org), bzw. Installation via ppm.

### 2.2.15.2 Oracle®

- PL/SQL muss installiert und aktiviert sein.

### 2.2.15.3 MS-SQL®

- Mindestversion MS-SQL Server 2005®, da die Unterstützung von `EXCEPT` und `INTERSECT` benötigt wird.

### 2.2.15.4 MySQL

- Mindestversion von MySQL ist 5.1.47. Ab dieser Version wurde ein bekanntes Deadlock-Problem gelöst.



### Wichtig:

*Für diese Version von MySQL gibt es kein MySQLmax mehr.*



### Wichtig:

*Damit **imperia 9** fehlerfrei auf die Datenbank zugreifen kann ist es notwendig, dass das Datenbank „Isolation Level“ auf `READ COMMITTED` gesetzt wird. **imperia** erledigt dies automatisch. Sollten jedoch noch andere Anwendungen auf die Datenbank zugreifen, kann dies zu Problemen führen. Voraussetzung dafür ist, dass der **imperia**-mysql-Benutzer das Privileg 'SUPER' besitzt. Dies kann mit dem Kommando*

```
GRANT SUPER on . to imperia
```

*zugewiesen werden.*

*Weitere Informationen können unter folgendem Link nachgelesen werden:*

<http://dev.mysql.com/doc/refman/5.0/en/set-transaction.html>

## 2.2.15.5 PostgreSQL

Die benötigte Mindestversion von PostgreSQL ist 8.1. Wir empfehlen aber die stabile Version 8.4.3-1 (zum Erstellungszeitpunkt dieser Dokumentation).



### Hinweis:

*Die Version 8.4.1-1 hat große Fehler und sollte auf keinen Fall eingesetzt werden.*

PI/PGSQL muss installiert und aktiviert sein.

## 2.2.15.6 SQLite

- SQLite sollte in der Version 1.29 installiert sein. Vorherige Versionen führen zu extrem langen Bearbeitungszeiten bei größeren Datenbankoperationen. (siehe auch Abschnitt 2.2.7 **Perl-Modul DBD::SQLite** auf Seite 6).

## 2.2.16 Weitere Informationen

Halten Sie während der Installation folgende Informationen bereit:

- Document Root des Webservers
- IP-Adresse
- URL
- CGI-BIN-Verzeichnis
- User-ID des Webservers (nur UNIX®)

Bei gravierenden Installations-Problemen wenden Sie sich bitte über das folgende Supportformular an den **imperia** -Support:

<https://servicecenter.imperia.net/servicecenter/support/anfragen/index.html.de>

Sollten Sie noch keine Zugangsdaten zum **imperia** -Servicecenter besitzen, wenden Sie sich bitte an [<support@imperia-ag.de>](mailto:support@imperia-ag.de), um diese zu erhalten.

## 2.3 Installation unter UNIX®-Derivaten

Zur Durchführung der Installation unter UNIX®-Derivaten gehen Sie wie folgt vor:

1. Achten Sie darauf, dass alle in Abschnitt 2.2 **Voraussetzungen** auf Seite 3 beschriebenen Voraussetzungen geschaffen wurden.
2. Falls Perl nicht in das Verzeichnis `/usr/local/bin` installiert wurde, legen Sie einen symbolischen Link von `/usr/bin/perl` nach `/usr/local/bin/perl` an.
3. Finden Sie heraus, unter welcher User-ID der Webserver läuft. Folgende Befehle liefern Hinweise:

```
cat /etc/passwd | grep -i www  
ypcat passwd | grep -i -E '(www|web)'
```

Überprüfen Sie gegebenenfalls auch die Konfigurationsdatei Ihres Webservers. Wenn Sie einen Apache-Webserver verwenden, suchen Sie in dessen Konfigurationsdatei nach den Begriffen `User` und `Group`:

```
egrep -i '(User|Group)' /path/to/httpd.conf
```

4. Nehmen Sie die in Schritt 3 gefundene User-ID an:

```
su wwwrun
```

5. Entpacken Sie das gelieferte Archiv in ein temporäres Verzeichnis.
6. Wechseln Sie in dieses Verzeichnis und starten Sie das Skript `install.pl`:

```
cd /Ihr/temporaeres/Verzeichnis
./install.pl
```

Es startet die interaktive Installations-Routine.

Eine Übersicht über alle möglichen Installations-Optionen erhalten Sie am Anfang von Kapitel 1 dieses Buches oder über den folgenden Befehl:

```
perl install.pl -h
```

7. Wurde das Skript als **root** gestartet, müssen Sie zunächst die User-ID des Webservers angeben. Geben Sie dann an, ob Sie ein Produktionssystem (Develop) oder ein Zielsystem (Live) installieren wollen.  
Geben Sie nacheinander die einzelnen Verzeichnisse an, die verwendet werden sollen. Hierzu gehören das CGI-BIN-Verzeichnis (optional), das SITE-Verzeichnis und das Document-Root-Verzeichnis. Aus Sicherheitsgründen sollte das SITE-Verzeichnis außerhalb des Document-Root-Verzeichnisses liegen, um unerwünschte Zugriffe zu verhindern. Ausgehend vom Ursprungsverzeichnis Ihrer Installation werden nicht existierende Verzeichnisse angelegt.
8. Nachdem alle Fragen beantwortet wurden, erhalten Sie eine Übersicht über die getätigten Angaben. Überprüfen Sie diese und bestätigen Sie sie mit der Eingabe „yes“.  
Möchten Sie einen Fehler korrigieren, geben Sie „no“ ein und wiederholen Sie die Eingaben.
9. Nachdem Sie Ihre Eingaben bestätigt haben wird **imperia** in den durch Sie angegebenen Verzeichnisse installiert.
10. Das temporäre Verzeichnis sollte erst nach der Installation eines Zielsystems gelöscht werden. Lesen Sie hierzu Abschnitt 2.5 **Installation eines Zielsystems** auf Seite 11.

## 2.4 Installation unter Windows® 2000 / 2008 Server / XP / Vista / 7

Um **imperia** unter Windows® 2000 / 2003 Server / XP / Vista / 7 zu installieren, gehen Sie wie folgt vor:

1. Achten Sie darauf, dass alle in Abschnitt 2.2 **Voraussetzungen** auf Seite 3 beschriebenen Voraussetzungen geschaffen wurden.
2. Entpacken Sie das gelieferte Archiv in ein temporäres Verzeichnis.
3. Starten Sie eine DOS-Eingabeaufforderung und wechseln Sie in das temporäre Verzeichnis.
4. Starten Sie das Installationsskript wie folgt:

```
perl install.pl
```

Eine Übersicht über alle möglichen Installationsoptionen erhalten Sie am Anfang dieses Buchs oder über den folgenden Befehl:

```
perl install.pl -h
```

5. Geben Sie an, ob Sie ein Produktionssystem (Develop) oder ein Zielsystem (Live) installieren wollen.

Geben Sie nacheinander die einzelnen Verzeichnisse an, die verwendet werden sollen. Hierzu gehören das CGI-BIN-Verzeichnis, das SITE-Verzeichnis und das Document-Root-Verzeichnis. Aus Sicherheitsgründen sollte das SITE-Verzeichnis außerhalb des Document-Root-Verzeichnisses liegen, um unerwünschte Zugriffe zu verhindern. Die Pfadangaben müssen mit Schrägstrichen (/) angegeben werden; andere Pfadangaben werden nicht verarbeitet. Verzeichnisse, die nicht existieren, werden, ausgehend vom Ursprungs-Verzeichnis, automatisch angelegt.

Geben Sie den URI der `cgi-bin`-Location Ihres Webservers an.

6. Nachdem alle Fragen beantwortet wurden, erhalten Sie eine Übersicht über die getätigten Angaben. Überprüfen Sie diese und bestätigen Sie sie mit der Eingabe „yes“.  
Wenn Sie einen Fehler in Ihren Angaben korrigieren möchten, geben Sie „no“ ein und korrigieren Sie die Eingaben.
7. Nachdem Sie Ihre Eingaben bestätigt haben, wird **imperia** in den durch Sie angegebenen Verzeichnisse installiert.
8. Das temporäre Verzeichnis sollte erst nach der Installation eines Zielsystems gelöscht werden.

## 2.5 Installation eines Zielsystems

Um ein Zielsystem zu installieren, verfahren Sie genauso wie bei der Installation eines Produktionssystems. Lediglich bei der Auswahl des Servertyps (DEVELOP oder LIVE) geben Sie „LIVE“ ein.

Im weiteren Verlauf der Installation müssen die entsprechenden Verzeichnisse für das Zielsystem angegeben werden. Verzeichnisse, die nicht existieren, legt das Skript, ausgehend vom Ursprungs-Verzeichnis, automatisch an.

## 2.6 Installieren eines Patches ("Hotfix")



### Achtung:

*Vor dem Einspielen eines Patches/Updates sollten Sie unbedingt ein Backup Ihres bestehenden **imperia**-Systems durchführen.*

Es ist besser das Update eines laufenden **imperia**-Systems durch das Einspielen eines Patches als durch eine komplette Neu-Installation durchzuführen.

Ein **imperia**-Hotfix umfasst genau die Dateien, die für die neue Version geändert wurden. Hotfixe beinhalten jedoch nur Änderungen bzw. Bugfixes bereits vorhandener Funktionen und Features. Neue Features werden nicht hinzugefügt.

Das Installationskript überschreibt weder Daten noch vorhandene Konfigurationsdateien. Im Zweifelsfall werden vorhandene Dateien mit einer anderen Endung (.001, .002, .003, ...) gesichert.

Es werden jedoch ALLE Dateien überschrieben, die eine der folgenden Endungen haben:

.pl, .pm, .pod, .txt, .jar, .js, .de, .uk, .gif, .jpg, .jpeg, .tif, .tiff, .psd, .png, .pdf, .rtf, .doc, .xls, .bat, .sh, .exe

Wenn Sie in einer dieser Dateien Änderungen gemacht haben, besteht die Möglichkeit, diese Datei vom Skript sichern zu lassen. Starten Sie dazu das Installations-Skript mit der Option `-k`. Geänderte Dateien werden dann vor dem Kopieren umbenannt und im Installationsprotokoll aufgelistet.

Während des Updates werden vorhandene Dateien mit den neuen Dateien verglichen, wodurch erreicht wird, dass nach dem Update eine genau definierte **imperia**-Version auf Ihrem System vorhanden ist. Das Installationsprotokoll sagt Ihnen exakt, welche Dateien ersetzt wurden. Sichern Sie das Installationsprotokoll und halten Sie es für eventuelle Supportanfragen bereit.

Der Eintrag "skipping" im Installationsprotokoll bzw. während der Installation zeigt, dass die betreffende Datei auf dem neuesten Stand ist und daher nicht ersetzt werden muss.

Um einen Patch bzw. ein Update zu installieren, gehen Sie wie folgt vor:

1. Führen Sie ein Backup Ihres bestehenden **imperia**-Systems durch.

2. Entpacken Sie das Archiv, in dem das Update enthalten ist, in ein temporäres Verzeichnis.
3. Starten Sie in diesem Verzeichnis das Skript `install.pl`.
4. Wenn Sie den Patch unter UNIX® installieren, beachten Sie die Hinweise in Bezug auf die Zugriffsrechte für Dateien in Abschnitt 2.3 **Installation unter UNIX®-Derivaten** auf Seite 9.
5. Folgen Sie den Anweisungen des Skripts.

## 2.7 Update von einer älteren imperia 9 -Version



### Achtung:

*Vor dem Update sollten Sie unbedingt ein Backup Ihres bestehenden **imperia** -Systems durchführen.*

Wenn Sie von einer älteren Version von **imperia** auf die aktuelle Version updaten möchten, müssen die folgenden Dinge besonders beachtet werden.

### 2.7.1 Updaten der Datenbankstruktur

Bitte führen Sie nach dem Update auf die aktuelle **imperia** -Version ein Update der Datenbankstruktur durch.

```
site/bin/site_doctree_admin.pl -U
```

Weitere Informationen hierzu finden Sie im Kapitel Abschnitt 2.10 **Smart Database Update** auf Seite 15

## 2.8 Konfiguration von imperia

**imperia** wird mit Hilfe verschiedener Parameter in der Datei `/site/config/system.conf` konfiguriert. Im Folgenden finden Sie eine Liste der wichtigsten Variablen und deren Bedeutung.

Alle Einzelheiten hierzu entnehmen Sie bitte dem Kapitel **Variablen zur System-Konfiguration** [`./admin/admin.pdf#adminConf`] im Administrationshandbuch.

### 2.8.1 ALLOW\_TEMPLATE\_FORM

Erlaubt Formulare in Templates. Mögliche Einstellungen sind:

- 1: Das `<form>`-Tag kann in Templates verwendet werden.
- 0: Das `<form>`-Tag kann in Templates nicht verwendet werden.

### 2.8.2 ALLOW\_TEMPLATE\_SCRIPT

Erlaubt Plug-Ins und Skripte in Templates (z.B. JavaScript oder Applets). Mögliche Einstellungen sind:

- 1: Plug-Ins und Skripte können in Templates verwendet werden.
- 0: Plug-Ins und Skripte können nicht in Templates verwendet werden.

### 2.8.3 ALLOW\_TEMPLATE\_FRAME

Erlaubt Frames und Layer in Templates. Mögliche Einstellungen sind:

- 1: Frames und Layer sind erlaubt.
- 0: Frames und Layer sind nicht erlaubt.

### 2.8.4 UPLOAD\_OBJECTS

Mit Hilfe dieses Parameters kann eingestellt werden, wie **imperia** verlinkte Objekte beim Freischalten behandelt. Mögliche Einstellungen sind:

## ALL

Es werden alle mit einem Dokument verlinkten Objekte freigeschaltet, die innerhalb der Attribute `src`, `background` oder `href` innerhalb der Tags `img`, `td`, `th`, `table`, `body`, `link`, `input` und `a` erscheinen.

## LINKS

Diese Einstellung entspricht der Einstellung „ALL“ mit dem Unterschied, dass hierbei das Anker-Tag (`<a>`) nicht berücksichtigt wird.

## Kein Eintrag

Es werden alle mit einem Dokument verlinkten Objekte freigeschaltet, die innerhalb der Attribute `src` oder `background` innerhalb der Tags `img`, `td`, `th`, `table`, `body` erscheinen.

## 2.8.5 OPERATING-SYSTEM

Dieser Parameter beschreibt die Art der Plattform, auf der **imperia** installiert ist. Mögliche Einstellungen sind:

- "DOS"
- "UNIX"

## 2.8.6 CGI-DIR

Dieser Parameter enthält den absoluten Pfad zum `CGI-BIN`-Verzeichnis des Produktionssystems, in dem alle CGI-Skripte von **imperia** liegen.

## 2.8.7 DOCUMENT-ROOT

In diesem Parameter wird der absolute Pfad zum Document-Root-Verzeichnis des Webservers gespeichert. Das Verzeichnis `imperia` sollte ein Unterverzeichnis dieses Verzeichnisses sein (z.B.: `"/usr/httpd/dev_serv/htdocs/imperia"`).

## 2.8.8 CGI-BIN

In diesem Parameter wird der relative Pfad zum `CGI-BIN`-Verzeichnis des Webservers gespeichert.

## 2.8.9 DEMOMODE

Mit diesem Parameter wird der Demomodus ein- bzw. ausgeschaltet. Mögliche Einstellungen sind:

- 1: Demomodus ist aktiviert.
- 0: Demomodus ist nicht aktiviert.

## 2.8.10 SERVER-SCHEME

Dieser Parameter enthält das Protokoll-Schema, entweder HTTP oder HTTPS (HTTP mit SSL Verschlüsselung).



### Hinweis:

*Diese Variable müssen Sie nicht selbst setzen. Vielmehr liest **imperia** die Einstellung aus der Umgebung des Webservers und speichert Sie in der Variablen ab.*

## 2.8.11 SERVER\_NAME

Dieser Parameter enthält den FQDN (Fully Qualified Domain Name) des Servers.



### Hinweis:

*Diese Variable wird automatisch gesetzt (siehe Abschnitt 2.8.10 **SERVER-SCHEME** auf Seite 13).*

## 2.8.12 SERVER-PORT

Dieser Parameter enthält den Port, auf dem der Server läuft. Die Standardeinstellung ist 80 für HTTP und 443 für HTTPS (HTTP mit SSL-Verschlüsselung).



### Hinweis:

*Diese Variable wird automatisch gesetzt (siehe Abschnitt 2.8.10 **SERVER-SCHEME** auf Seite 13).*

## 2.8.13 HTTP\_HOST

Dieser Parameter enthält das Verzeichnis, das via HTTP als Document Root angesprochen wird. Standardeinstellung ist "/".

## 2.8.14 LOCK\_EX, LOCK\_SH, LOCK\_UN, LOCK\_NB

Zusätzliche Parameter zum File Locking.

## 2.8.15 COOKIE\_TTL

Dieser Parameter enthält den Gültigkeitszeitraum (Time To Live) einer Usersession. Bleibt eine Usersession über diesen Zeitraum hinaus ungenutzt (idle), wird sie automatisch beendet und der User muss sich erneut anmelden.

## 2.8.16 SAFE\_TEMPLATE

Wird dieser Parameter auf den Wert 1 gesetzt, dann wird der Perl-Code im Template oder in Flexmodulen in einer sicheren Umgebung ausgeführt.

## 2.8.17 REG\_KEY, REG\_KEY2, REG\_NAME

Diese Parameter enthalten die Lizenzschlüssel sowie den Kundennamen.

## 2.8.18 IMPERIA\_ERROR\_LOG

Mit diesem Parameter lenken Sie die Ausgabe auf STDERR in eine Datei um. Diese Datei befindet sich im Verzeichnis site/logs Ihrer **imperia**-Installation und heißt error.log. Insbesondere bei der Verwendung eines Apache Webservers unter Windows® 2000 kann es zu Abstürzen kommen, wenn mehr als 1KB Daten nach STDERR beschrieben werden. Wird STDERR in eine Datei umgeleitet, werden diese Abstürze vermieden. Mögliche Werte sind 1 bzw. eine Zeichenkette für die Aktivierung der Umlenkung oder 0 für deren Deaktivierung.

## 2.8.19 LATCHING

Normalerweise verlassen Sie den Bearbeiten-Schritt eines Dokuments beim Abspeichern Ihrer Eingaben automatisch. Das System führt Sie dann auf den Schreibtisch. Sollten Sie eine Möglichkeit zum Speichern Ihrer Eingaben wünschen, ohne den Bearbeiten-Schritt verlassen zu müssen, können Sie die Speicherung mit der Variablen `LATCHING` aktivieren. Wenn Sie diese auf 1 setzen, erscheint im Bearbeiten-Schritt der Button **Zwischenspeichern**, mit dem Sie Ihre Eingaben abspeichern und das Dokument weiterhin bearbeiten können.

## 2.9 Modul-Installation

### 2.9.1 Verwendung des Grids

Beim Grid handelt es sich um die Oberfläche zum Erstellen und Ändern von Workflows. Die client-seitigen Voraussetzungen für die Verwendung des Grids sind:

- ein JavaScript-fähiger Browser, der außerdem den Tag `embed` oder `object` darstellen kann.

- Das JavaRuntime-Plug-In in der aktuellen Version; erhältlich für die einzelnen Plattformen unter <http://java.sun.com/>.



**Hinweis:**

*openJDK (IcedTea Java) wird derzeit nicht unterstützt.*

## 2.9.2 OneClickEdit installieren

OneClickEdit, kurz OCE, gibt Ihnen die Möglichkeit, beim Browsen auf dem Zielsystem im Internet oder Intranet die gerade besuchte Seite direkt in **imperiam** zu bearbeiten. OneClickEdit ist eine Browsererweiterung, die Sie nachträglich auf Ihrem Rechner installieren müssen. Sollten Sie nicht berechtigt sein, Browsererweiterungen auf Ihrem Rechner zu installieren, sprechen Sie bitte mit Ihrem Systemadministrator.

OCE unterstützt folgende Browser:

- Microsoft Internet Explorer für Windows 2000, 2003, XP ab Version 5.5
- Netscape ab Version 7
- Mozilla (Mozilla.org) ab Version 1.0
- (Mozilla) Firefox ab Version 1.5 (Windows), bzw. 1.0.7 (Linux)

Für die verschiedenen Browser stehen Ihnen jeweils eigene Installationspakete zur Verfügung. Um die Erweiterung zu installieren, klicken Sie auf den Button **OneClickEdit** in dem Abschnitt, der zu Ihrem Browser passt. Es wird ein Dialog geöffnet, in dem Sie wählen können, ob das Installationsprogramm direkt ausgeführt oder gespeichert werden soll. Wählen Sie eine der Optionen.

## 2.10 Smart Database Update



**Achtung:**

*Vor dem Update sollten Sie unbedingt ein Backup Ihres bestehenden **imperiam** -Systems durchführen.*

Seit der Version **imperiam 8.5.4** verwendet die SQL-Datenhaltung in **imperiam** mehr Constraints um die Datenbankintegrität besser sicherstellen zu können. Dies wurde mit **imperiam 9** weiter ausgebaut. Die Indexe im Datenbankschema wurden überarbeitet, um mehr Leistung zu erzielen. Diese Indexe werden nicht automatisch aktualisiert. Sehen Sie hierfür bitte im Kapitel 'Upgrade und Wartung' nach. Vor **imperiam 8.5.4** war das Updaten des Datenbankschemas ein mühseliger Prozess. Die Dokumentenhaltung musste zuerst nach File65 umgewandelt werden. Anschließend wurde eine leere Datenhaltung mit dem neuen Datenbankschema erstellt, welche zu guter Letzt wieder mit den vorher kopierten Daten gefüllt werden musste. Da dieser Vorgang sehr zeitintensiv war, konnte es zu beachtlichen Ausfallzeiten kommen. Zusätzlich verursachte der gesamte Prozess eine erhebliche Auslastung. Mit der Zeit verbesserte der 'batch-mode' des clone\_repository.pl Skripts diese Situation ein wenig, doch war eine erforderliche Ausfallzeit unumgänglich.

### 2.10.1 Update und Wartung

Seit **imperiam 8.5.4** werden zwei neue Skripte mitgeliefert, die das Upgraden und die Wartung der Datenhaltung erheblich erleichtern.

#### 2.10.1.1 site\_doctree\_admin.pl

In **imperiam 8.5.4** wurde das Skript `site/bin/site_doctree_admin.pl` hinzugefügt, das die Datenbankmigration in Echtzeit durchführt. Wann immer es möglich ist, teilt das Skript den Vorgang in kleine Einzelschritte auf. So kann der Vorgang unterbrochen und später an der gleichen Stelle fortgesetzt werden.



**Hinweis:**

*Bitte beachten Sie, dass nur die Datenhaltungen PostgreSQL und SQLite ein Rollback für Änderungen in der Datenbankstruktur (Tabellen hinzufügen/löschen, Constraints, etc.) unterstützen.*

Für die Datenhaltungen Oracle und SQLite gibt es ein paar wenige Bedingungen. Somit kann eine unterbrochene Migration evtl. den manuellen Eingriff in den Prozess erfordern.

Wenn Sie das Skript ohne jegliche Parameter aufrufen, analysiert dieses die aktuelle Version des Datenbankschemas und schlägt evtl. ein Update vor.

Mit dem Aufruf:

```
site_doctree_admin.pl --initialize
```

starten Sie die Migration der Datenbank. Dieser Aufruf kann auch mit der Option „--dry-run“ kombiniert werden. Das führt dazu, dass keine wirklichen Änderungen durchgeführt werden, sondern lediglich eine Migration simuliert wird.

Alle verfügbaren Parameter erhalten Sie mit:

```
site_doctree_admin.pl --help
```

### 2.10.12 site\_doctree\_repair.pl

Wenn Sie die Änderungen durchführen, kann es in einzelnen Fällen zu Problemen mit doppelt vorhandenen Schlüsseln kommen. Für diesen Fall steht das Skript `site_doctree_repair.pl` zur Verfügung, welches die doppelten Zeilen aus der Datenbank entfernt. Wenn der Aufruf von

```
site_doctree_admin.pl --initialize
```

fehlschlägt, können Sie in der Ausgabe des Skripts nachsehen, welche Tabelle den Fehler verursacht hat. Wenn Sie erst einmal die betroffene Tabelle kennen, können Sie diese mit `site_doctree_repair.pl` reparieren:

```
site_doctree_repair.pl --table=TABLE --repair
```

Auch diesen Aufruf können Sie zuerst mit dem Parameter `--dry-run` kombinieren um zu sehen, was das Skript für Änderungen an der jeweiligen Tabelle vornehmen wird. Alle verfügbaren Parameter können Sie mit

```
site_doctree_repair.pl --help
```

nachsehen.

### 2.10.2 Spachunterstützung im Betriebssystem

Sollte die Übersetzung in der Oberfläche nicht funktionieren, so kann dies daran liegen, dass die Spachunterstützung im System nicht installiert ist. Dies kann unter Linux-Systemen mit dem Kommando

```
locale -a
```

geprüft werden.

Sollte die entsprechende Sprache (hier im Beispiel deutsch (`de_DE.utf8`)) nicht installiert sein, so müssen Sie sie nachträglich installieren. Auf DEBIAN-Systemen funktioniert dies z.B. mit:

```
dpkg-reconfigure locales
```

Auf SuSE-Systemen können Sie die Sprachen über das Yast auswählen.

## Kapitel 3. Migration <sup>19+</sup>

### 3.1 Vorbereitung

#### Update einer bestehenden imperia 7.5 Installation



##### Hinweis:

*Bitte beachten Sie, dass mit dem Erscheinen von **imperia 9** keine **imperia 7** Versionen mehr supportet werden.*

1. Um **imperia 7.5** auf **imperia 9** upzudaten, müssen Sie vorher ein Update auf **imperia 8.6** durchführen. Bitte lesen Sie in der Dokumentation von **imperia 8.6** die entsprechenden Hinweise zum Update nach.
2. Sollte Ihr System noch nicht mit dem UTF-8 Zeichensatz arbeiten, muss dieser bei der Umstellung auf **imperia 8.6** nun verwendet werden, da **imperia 8** intern nur noch mit UTF-8 arbeitet.
3. Stellen Sie, soweit noch nicht geschehen, in **imperia 8.6** ihre Medienverwaltung von der MBD (Mediendatenbank) auf das MAM (Media Asset Management) um. In **imperia 9** wird die MDB nicht weiter unterstützt.

#### Update einer bestehenden imperia 8 Installation

1. Um **imperia 8** auf **imperia 9** upzudaten, müssen Sie vorher ein Update auf **imperia 8.6** durchführen. Bitte lesen Sie in der Dokumentation von **imperia 8.6** die entsprechenden Hinweise zum Update.
2. Stellen Sie, soweit noch nicht geschehen, in **imperia 8.6** ihre Medienverwaltung von der MBD (Mediendatenbank) auf das MAM (Media Asset Management) um. In **imperia 9** wird die MDB nicht weiter unterstützt.
3. Bitte überprüfen Sie, dass bei den Zielsystemen keine Backslashes verwendet werden und dass am Ende der Pfadangaben kein Slash verwendet wird.

#### Allgemeine Vorbereitungen

1. Sollten Sie von einer älteren Version auf **imperia 8.6** aktualisiert haben, überprüfen Sie bitte noch einmal Ihr System, bevor Sie mit der Migration auf **imperia 9** fortfahren.
2. Lesen Sie bitte das Release-Dokument und die Installationsanleitung zu **imperia 9**.
3. Die Migration der Workflowschritte erfordert das Perlmodul XML::Parser in der Version 2.40. Bitte installieren Sie dies vor dem Start der Migration.
4. Erstellen Sie zur Sicherheit immer ein Backup von Ihrem System und der Datenbank.
5. Beenden Sie den Hermes auf dem Basissystem und auf den Livesystemen.
6. Stoppen Sie den Webserver.

### 3.2 Migrationsschritte

Für die im Folgenden aufgeführten Programme können immer mit dem Parameter „--help“ alle verfügbaren Optionen angezeigt werden.

Bei den meisten Programmen können über den Parameter -v (verbose) detailliertere Informationen ausgegeben werden. Dies führt zu einer etwas längeren Ausführung des Programmes.

Bei den aktuellen Programmen ist der Ablauf in zwei Abschnitte geteilt. Im ersten Schritt wird eine Liste mit Aufgaben (ToDo-Liste) erstellt, die im zweiten Schritt abgearbeitet wird. Vorteil ist, dass der zweite Teil beliebig oft unterbrochen und neu gestartet werden kann.

### 3.2.1 Datenhaltung File65

1. Installation von **imperia 9** auf dem Basissystem.

```
./install.pl -s /srv/www/imperia/dev/imperia.de/site -c none
```

Die im Kommando verwendeten Parameter haben folgende Bedeutung:

-c none                    Mit diesem Parameter kann das cgi-Verzeichnis für die Installation übergeben werden. In einer mod\_perl- Umgebung kann auf den Einsatz von cgi-Scripten verzichtet werden. In diesem Fall wird der Parameter auf „none“ gesetzt.

**Tabelle 3.1.**

Bitte ersetzen Sie den Pfad `/srv/www/imperia/dev/imperia.de/site` im Beispiel durch Ihre Pfadangabe.

2. Installation von **imperia 9** auf allen Livesystemen.

```
./install.pl -l -s /srv/www/imperia/live/imperia.de/site
```

Bitte ersetzen Sie den Pfad `/srv/www/imperia/dev/imperia.de/site` im Beispiel durch Ihre Pfadangabe.

3. Aktualisieren des Datenbankschemas.

```
./imperia.de/site/bin/site_doctree_admin.pl -U
```

Die im Kommando verwendeten Parameter haben folgende Bedeutung:

-U (update)                Update des Datenbankschemas.

**Tabelle 3.2.**

4. Migration der Zielsysteme, der freigeschalteten und freizuschaltenden Dokumente.

```
./imperia.de/site/bin/site_migrate_9_file65.pl -t -b
```

Die im Kommando verwendeten Parameter haben folgende Bedeutung:

-t (--todo)                ToDo-Liste erstellen.

-b (--batch)                ToDo-Liste abarbeiten. Kann unterbrochen und neu gestartet werden.

**Tabelle 3.3.**

Kann ggf. lange laufen.

5. Migration der Templates und der Rollen.

```
./imperia.de/site/bin/site_migrate_9.pl -a -f
```

-a (--all)                Führt alle möglichen Migrationen auf einmal durch.

-f (--force)                Bestätigungsnachfragen überspringen.

**Tabelle 3.4.**

6. Referenzen in **imperia 9** aktualisieren.

```
./imperia.de/site/bin/site_rebuild_db.pl -t -b
```

Die im Kommando verwendeten Parameter haben folgende Bedeutung:

|              |  |
|--------------|--|
| -t (--todo)  | ToDo-Liste erstellen.  |
| -b (--batch) | ToDo-Liste abarbeiten. Kann unterbrochen und neu gestartet werden. |

**Tabelle 3.5.**

Kann ggf. sehr lange laufen.

### 3.2.2 Datenhaltung SQL-DB

1. Installation von **imperia 9** auf dem Basissystem.

```
./install.pl -s /srv/www/imperia/dev/imperia.de/site -c none
```

Die im Kommando verwendeten Parameter haben folgende Bedeutung:

|         |   |
|---------|---|
| -c none | Mit diesem Parameter kann das cgi-Verzeichnis für die Installation übergeben werden. In einer mod_perl- Umgebung kann auf den Einsatz von cgi-Scripten verzichtet werden. In diesem Fall wird der Parameter auf „none“ gesetzt. |
|---------|---|

**Tabelle 3.6.**

Bitte ersetzen Sie den Pfad `/srv/www/imperia/dev/imperia.de/site` im Beispiel durch Ihre Pfadangabe.

2. Installation von **imperia 9** auf allen Livesystemen.

```
./install.pl -l -s /srv/www/imperia/live/imperia.de/site
```

Bitte ersetzen Sie den Pfad `/srv/www/imperia/dev/imperia.de/site` im Beispiel durch Ihre Pfadangabe.

3. Aktualisieren des Datenbankschemas. Migration der Zielsysteme, der freigeschalteten und freizuschaltenden Dokumente.

```
./imperia.de/site/bin/site_doctree_admin.pl -U
```

Die im Kommando verwendeten Parameter haben folgende Bedeutung:

|             |                              |
|-------------|------------------------------|
| -U (update) | Update des Datenbankschemas. |
|-------------|------------------------------|

**Tabelle 3.7.**

Kann ggf. lange laufen.

4. Migration der Templates, Zielsysteme und der Rollen.

```
./imperia.de/site/bin/site_migrate_9.pl -a -f
```

|              |  |
|--------------|--|
| -a (--all)   | Führt alle möglichen Migrationen auf einmal durch. |
| -f (--force) | Bestätigungsnachfragen überspringen.               |

**Tabelle 3.8.**

5. Datenbankstruktur optimieren.

```
./imperia.de/site/bin/site_doctree_admin.pl -V
```

Die im Kommando verwendeten Parameter haben folgende Bedeutung:

-V (vacuum) Datenbankstruktur optimieren.

**Tabelle 3.9.**

Kann ggf. lange laufen.

6. Referenzen in **imperia 9** aktualisieren.

```
./imperia.de/site/bin/site_rebuild_db.pl -t -b
```

Die im Kommando verwendeten Parameter haben folgende Bedeutung:

-t (--todo) ToDo-Liste erstellen.

-b (--batch) ToDo-Liste abarbeiten. Kann unterbrochen und neu gestartet werden.

**Tabelle 3.10.**

Kann ggf. sehr lange laufen.

## 3.3 Nachbereitung

### Manuelle Nachbereitungen in **imperia 9**

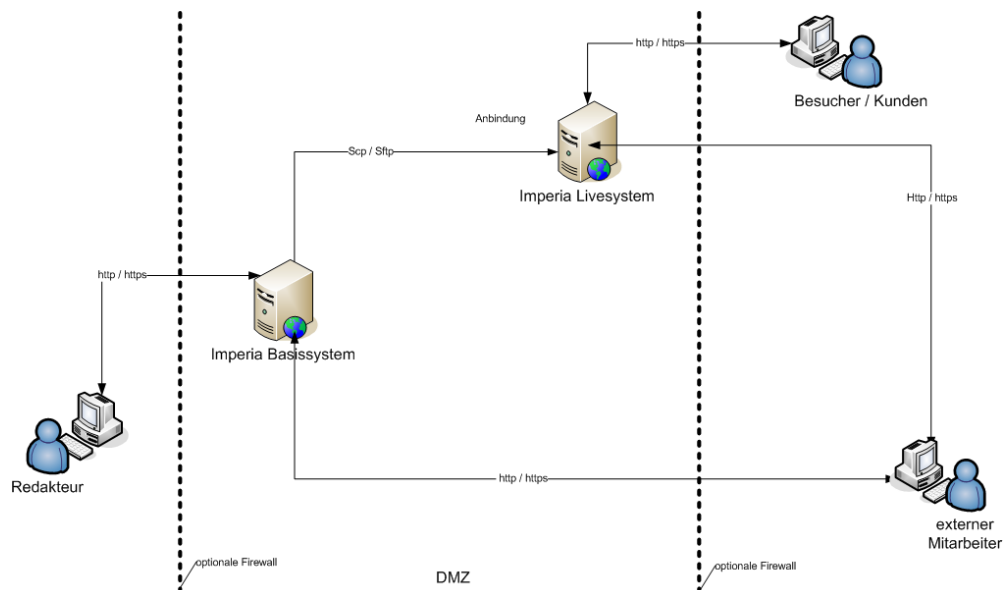
1. Werden bei der Ausführung von `site/bin/site_migrate_9.pl` anonyme Variablen in den Templates gefunden, so werden diese angezeigt. Diese anonymen Variablen müssen manuell ersetzt werden.
2. Anlegen der eigenen Menüs

### Allgemeine Nachbereitung

1. Aktualisieren Sie die Konfigurationsdatei des Webservers (`cgi` oder `mod_perl`). Lesen Sie dazu bitte die Details in Abschnitt 4.1 **Ausführungsschicht** auf Seite 23.
2. Starten Sie den Webserver wieder.
3. Starten Sie den Hermes auf dem Basissystem und auf den Livesystemen wieder.
4. Melden Sie sich am System an und aktualisieren Sie Ihren Lizenz-Schlüssel. Anschließend müssen Sie in einer `mod_perl` Umgebung den Webserver einmal neu starten, da die Lizenzinformationen gecached werden.

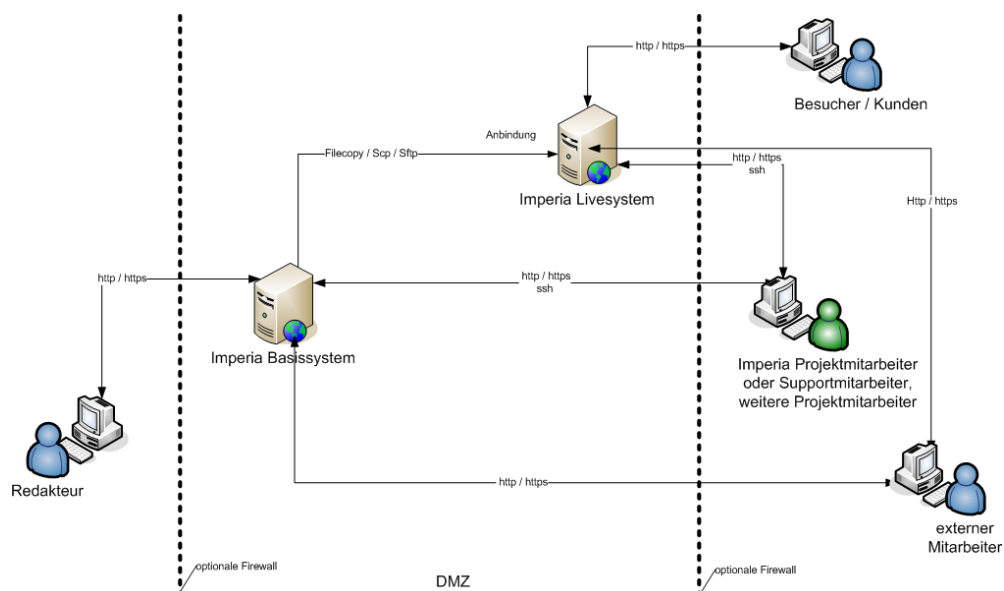
## Kapitel 4. Betriebskonzept 19+

Das **imperia** -System basiert auf einer Staging-Architektur. Das Basissystem, auch Redaktionsserver oder Entwicklungssystem genannt, wird vor allem von den Autoren genutzt, um die Inhalte zu managen. Davon — zumeist auch physikalisch — getrennt läuft die Liveumgebung von **imperia**, in die die Inhalte publiziert werden.



Bei allen Überlegungen zum technischen Betrieb von **imperia** muss berücksichtigt werden, ob und wie Partner, Dienstleister und die Imperia AG Zugriff auf die Systeme haben oder haben sollen.

Es kann außerdem in Projekten oder Supportfällen hilfreich sein, wenn die Imperia AG Zugriff auf die Kommandozeile (z.B. ssh unter Unix oder redesktop/vnc unter Windows) sowie Zugriff auf die Webseiten des Redaktionssystems und das **imperia** Livesystem hat bzw. bekommen kann. So können Supportfälle schneller und problembezogen untersucht werden.



Die folgenden Ausführungen beschreiben die möglichen produktiven Systemumgebungen. Testsysteme können, müssen aber nicht, komplex aufgebaut sein. Siehe dazu auch Abschnitt 4.7 **Testsysteme** auf Seite 34.

Das Entwicklungssystem wird je nach Ausrichtung der Publikation (Internet bzw. Intranet) entsprechend ausgerichtet. Bei Produktivsystemen wird normalerweise ein Server für das **imperia** -Develop System und ein weiterer für das **imperia** -Livesystem verwendet.

## 4.1 Ausführungsschicht

### 4.1.1 Apache mod\_perl

Das Apache-Modul mod\_perl ist ein in den Webserver Apache eingebetteter Perl-Interpreter. Dadurch muss der Perl-Interpreter nicht wie bei CGI bei jedem Aufruf neu gestartet werden. Über einen Handler werden alle Requests an die entsprechenden Controller weitergeleitet und somit die Seiten ausgeliefert.

Zusätzliche Apache2 Web-Server Konfiguration (z.B. vhost):

```
PerlOptions +Parent

PerlModule Apache2::RequestRec
PerlModule Apache2::compat

<Perl>
    unshift @INC, '/var/www/imperia/site/modules/core';
    unshift @INC, '/var/www/imperia/site/modules/collection';
    push @INC, '/var/www/imperia/site/modules/fallback';
</Perl>
<Location /imp/>
    SetHandler modperl
    PerlResponseHandler Imperia::Core::Backends::ModPerl20
</Location>
```

Ersetzen Sie `/var/www/imperia/site` mit dem absoluten Pfad zu Ihrem **imperia** System.

Wenn Sie das Prefix `"/imp"` in etwas anderes ändern müssen, dann müssen Sie auch die system.conf Variable `URI_PREFIX` entsprechend anpassen. Diese Variable darf keinen Slash enthalten. Hingegen in der `httpd.conf` macht es schon Sinn, den abschließenden Slash anzuhängen, so dass der Apache alle Anfragen an `"/imp"` an `"/imp/"` weiterleitet.



#### Hinweis:

*Bitte beachten Sie für den Betrieb unter Windows, dass es für die interne Kommunikation zwischen Webserver und Applikation notwendig ist, dass der Pfad zum Apache Webserver (z.B: C:\Apache2.2\bin ) in die Pfad Umgebungsvariable eingetragen wird und dass nur genau ein virtueller Host mod\_perl ausliefern kann.*



#### Hinweis:

*Bitte beachten Sie, dass es in Abhängigkeit der mod\_perl Version unter Umständen Probleme gibt, mehrere virtuelle Apache-Hosts parallel zu betreiben. Es ist daher in diesen Fällen ratsam, das System auf einer Apache-Einzelinstallation zu betreiben.*



#### Hinweis:

*Bei der Verwendung einer Oracle-Datenbank müssen die Oracle-Umgebungsvariablen in das Startup-Script des Apache- Webservers eingetragen werden, wenn mod\_perl verwendet werden soll.*

```
export NLS_LANG="AMERICAN_AMERICA.WE8ISO8859P15"
export NLS_NCHAR_CHARACTERSET="WE8ISO8859P15"
```

*Nur dadurch werden diese Variablen vor dem mod\_perl geladen. Jede andere Stelle kommt in der Ausführung zu spät und mod\_perl ist bereits geladen; die Umgebungsvariablen wirken sich nicht auf mod\_perl und imperia aus. Folge: Die Datenbank kann nicht richtig angesprochen werden. Bei CGI-Umgebungen reicht ein Eintrag in die default.conf in folgender Form:*

```
env_NLS_LANG = AMERICAN_AMERICA.WE8ISO8859P15
env_NLS_NCHAR_CHARACTERSET = WE8ISO8859P15
```

### 4.1.2 Fast-CGI

Beispielkonfigurationen für Fast-CGI:

Apache module mod\_fcgid

```
ScriptAlias /cgi-bin /srv/www/imperia/cgi-bin
<Location /cgi-bin/imperia-fcgi.pl>
    SetHandler fcgid-script
    Options ExecCGI
    Order allow,deny
    Allow from all
</Location>
```

Apache module mod\_fcgid

```
ScriptAlias /fcgi-bin /srv/www/imperia/fcgi-bin
<Location /fcgi-bin>
    SetHandler fcgid-script
    Options ExecCGI
    Order allow,deny
    Allow from all
</Location>
```

Apache module mod\_fastcgi

```
AppClass /srv/www/imperia/dev/cgi-bin/imperia-fcgi.pl
Alias /cgi-bin /srv/www/imperia/cgi-bin
<Location /imp/>
    SetHandler fastcgi-script
</Location>
```

Apache module mod\_fastcgi

```
ScriptAlias /cgi-bin /srv/www/imperia/cgi-bin
<Location /cgi-bin/imperia-fcgi.pl>
    SetHandler fastcgi-script
    Options ExecCGI
    Order allow,deny
    Allow from all
</Location>
```

### 4.1.3 Interner Webserver <sup>19+</sup>

Für Entwicklungsaufgaben kann statt einer mod\_perl Umgebung auch der interne Webserver site/bin/site\_imperia.pl verwendet werden. Insbesondere beim Debuggen ist er eine große Hilfe.

Die Konfiguration erfolgt in der Datei site/config/httpd.conf

Über den Aufruf `site/bin/site_imperia.pl --help` erhalten Sie die Hilfe zum interner Webserver:

```
-p, --port=PORT           the port to listen on, defaults to 2404
-i, --interface=INTERFACE comma separated list of interfaces that
                           should accept connections, defaults to
                           the loopback interface 127.0.0.1; use the
                           empty string in order to activate all
                           interfaces
-b, --background         run in the background
-t, --type=SERVER_TYPE  the base class of Net::Server to use as
                           the underlying server engine, defaults to
                           "Net::Server::Single"

-h, --help               display this help page and exit
-v, --verbose            print verbose debugging information
```

Die Verwendung des internen Web-Servers ist ganz einfach. Starten Sie einfach `site_imperia.pl` im Verzeichnis `site/bin` und spezifizieren Sie das Interface, dass Sie verwenden möchten. Mit dem Aufruf

```
./site_imperia.pl --interface=* -b
```

startet der interne Webserver und „lauscht“ auf allen Interfaces. Der Port ist per default 2404.

## 4.2 Speicherort

Das **imperia** Basissystem läuft im Webserver. Die verwalteten Dokumente können im Filesystem oder in einer Datenbank verwaltet werden. Aus ihnen werden jedoch spätestens beim Verlassen des Workflows Dokumente, die auch im Filesystem abgelegt werden, um auf die jeweiligen Zielsysteme transferiert zu werden. Zusätzlich liegen alle Module und Dateien zur Auslieferung der **imperia** -Oberfläche im Dateisystem. Dadurch nutzt das **imperia** -System das Filesystem recht intensiv.

### 4.2.1 Lokal

Die Nutzung des lokale Speichers, sprich der lokalen Fesplatte, ist immer eine gute Idee. Die Wege sind kurz und die Übertragung ist schnell.

### 4.2.2 SAN

Auch bei der Nutzung eines „quasi-“lokalen Speichers in einem SAN treten üblicherweise keine Probleme auf.

### 4.2.3 NAS

„Network Attached Storages“ sollten nur nach intensiver Prüfung betrieben werden, da in der Vergangenheit öfter Probleme beim Betrieb aufgetreten sind. Ursache dafür ist, dass diese Systeme dateibasierte Protokolle (wie NFS oder SMB/CIFS) verwenden. Diese Protokolle sind teilweise sehr langsam im Gegensatz zum lokalen Filesystem. Des Weiteren traten bei NFS (Versionen 2 und 3) Probleme auf, da diese Versionen kein „File-Locking“ unterstützten.

## 4.3 Datenbanken

### 4.3.1 PostgreSQL

Vorteile

- Die Software ist kostenlos.
- Standardisiert.

- Sehr einfache Installation und Administration über die Shell.

Nachteile

- Administration nur über die Shell.
- Nicht so viele Tools.

### 4.3.2 Oracle

Vorteile

- Gute Performance.
- Sehr verbreitet.
- Viele Funktionen.

Nachteile

- Software ist teuer.
- Höherer Administrationsaufwand.
- Teilweise Experten-Know-How notwendig.

### 4.3.3 MySQL

Vorteile

- Die Software ist kostenlos.
- Ist sehr verbreitet.
- Viele Tools.

Nachteile

- Oft instabile, inkompatible Releases,

### 4.3.4 MS-SQL

Vorteile

- Sinnvoll in homogenen MS-Umgebungen.

Nachteile

- Anbindung nur über ODBC!

## 4.4 Paradigmen

### 4.4.1 Entwicklungssystem

- Datenhaltung muss "schnell" sein.
- Für „kleine“ Projekte (weniger als 10.000 Dokumente) ist die Datenhaltung im Filesystem ideal.

- Nicht mehr benötigte Dokumente und Dokumentversionen sollten automatisiert aus dem System entfernt werden.
- Mandanten, die keine Verbindungen untereinander haben, sollten jeweils auf eigenen **imperia** -Instanzen umgesetzt werden.
- Die Netzwerkanbindung zwischen Autor und **imperia** -Server sollte performant sein.
- Zur Entkopplung sollte das Livesystem auf einer getrennten Hardware betrieben werden.

#### 4.4.2 Livesystem

Die statischen Livesysteme skalieren mit der Hardware. Sollte ein System die Last nicht mehr abdecken, können weitere parallele Systeme in Betrieb genommen werden, die dann als weitere Zielsysteme oder besser noch per `rsync` vom ersten System befüllt werden.

### 4.5 Standardsysteme

Das **imperia** Basissystem läuft auf jedem Betriebssystem für das eine Perl-Implementierung existiert. Üblicherweise werden

- Windows,
- Linux/Unix oder
- Solaris

verwendet.

Als Webserver eignen sich alle Webserver, für die eine `mod_perl`-Umgebung, eine CGI-Schnittstelle oder einer Fast-CGI-Schnittstelle zu Perl existiert. Hier wird meistens der „Apache Webserver“ eingesetzt. Aber auch

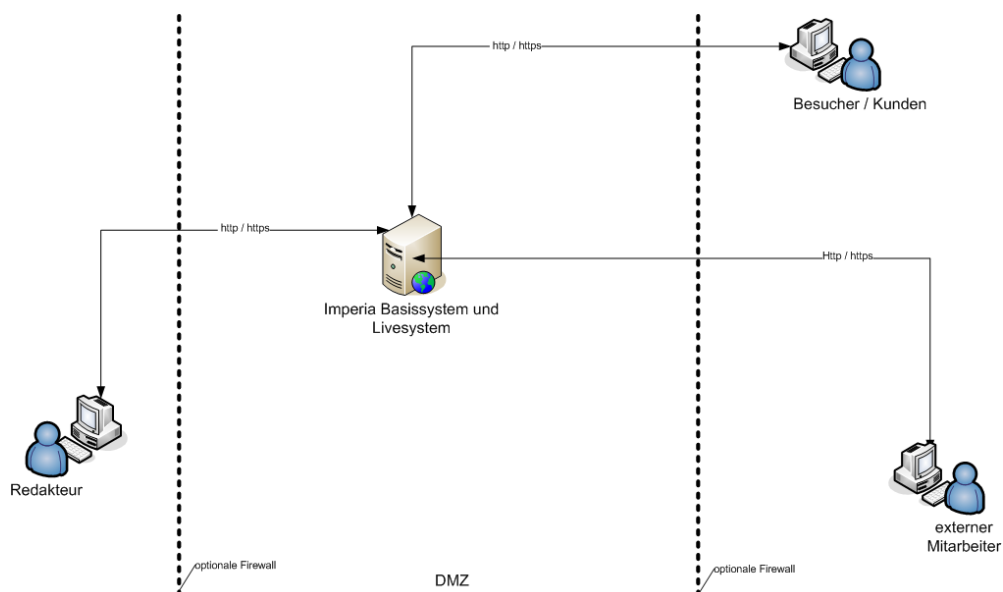
- IIS,
- IPlanet

können verwendet werden.

Neben der aktuellen Perl-Version muss eine zur Perl-Version passende Version von ImageMagick auf dem System installiert werden.

#### 4.5.1 Minimal

Diese Umgebung ist die minimale Grundkonfiguration.



|                 |                     |
|-----------------|---------------------|
| Prozessoren     | 2 CPUs              |
| Hauptspeicher   | 2GB RAM             |
| Plattenspeicher | 100GB ggf. als RAID |

**Tabelle 4.1. Konfiguration „Minimal-System“**

Als Datenhaltung wird das Filesystem verwendet.

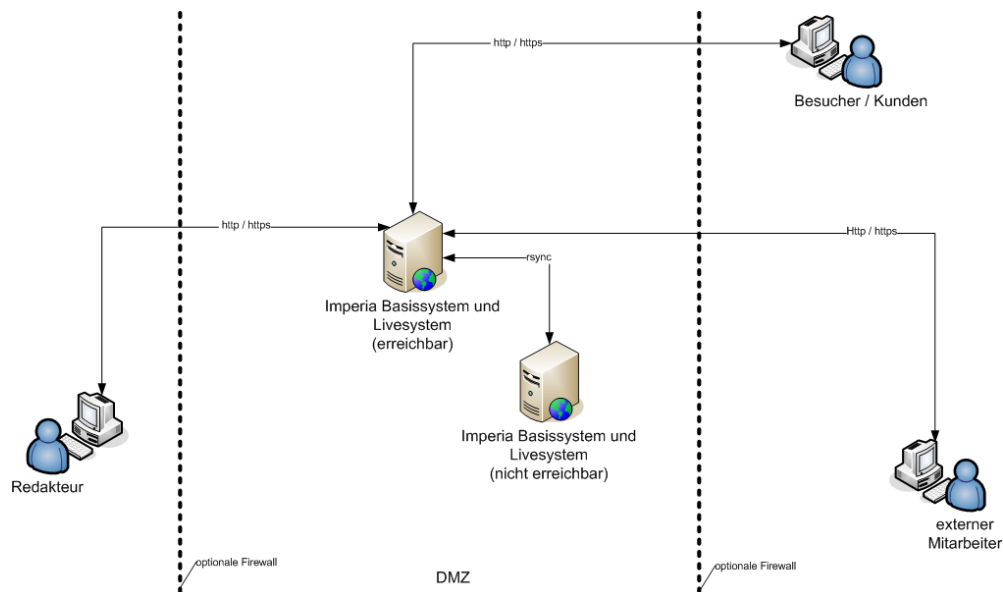
Durch die zwei Prozessoren ist sichergestellt, dass der Webserver und der **imperia** -Hintergrunddienst Hermes sich nicht gegenseitig in der Auslastung des Systems beeinflussen.

Eine solche Konfiguration wird meistens für kleinere Projekte oder Testsysteme eingesetzt.

In den meisten Internetumgebungen wird das **imperia** -System im Intranet aufgestellt und das **imperia** -Livesystem in der DMZ. Dadurch ist das **imperia** Develop System nur über das Intranet erreichbar.

### 4.5.2 Minimal Standby

Diese Umgebung ist die Grundkonfiguration für ein System mit einem redundanten Server.



|                 |                     |
|-----------------|---------------------|
| Prozessoren     | 2 CPUs              |
| Hauptspeicher   | 2GB RAM             |
| Plattenspeicher | 100GB ggf. als RAID |

**Tabelle 4.2. Konfiguration „Minimal Standbysystem“**

Das Standbysystem sollte analog zum führenden System aufgebaut sein.

Als Datenhaltung wird in dieser Umgebung das Filesystem verwendet.

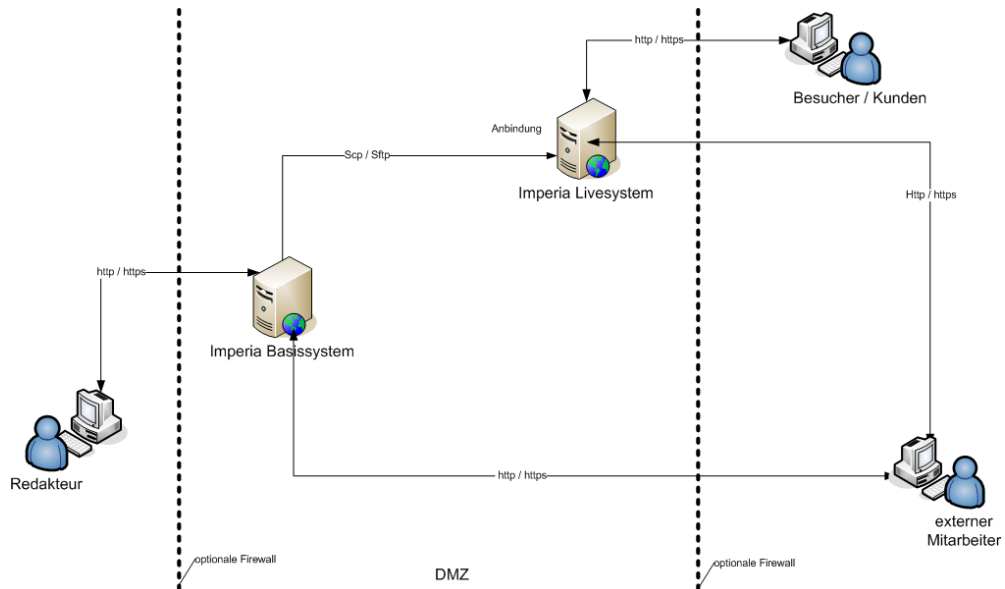
Beide Systeme sind aktiv, nur das Produktivsystem ist erreichbar. Es erfolgt ein regelmäßiger Abgleich der **imperia** -Systemkomponenten und der Bewegdaten auf Betriebssystemebene (z.B. mit rsync).

Im Falle eines Ausfalls des Produktivsystems muss der Standby-Server durch geeignete Maßnahmen auf Betriebssystemebene (IP-Umschaltung/Failover) erreichbar gemacht werden.

Eine solche Konfiguration wird meistens für kleinere Projekte oder Testsysteme eingesetzt, wobei Wert auf Ausfallsicherheit (innerhalb der Reaktionszeit) gelegt wird.

### 4.5.3 Standard Filesystem

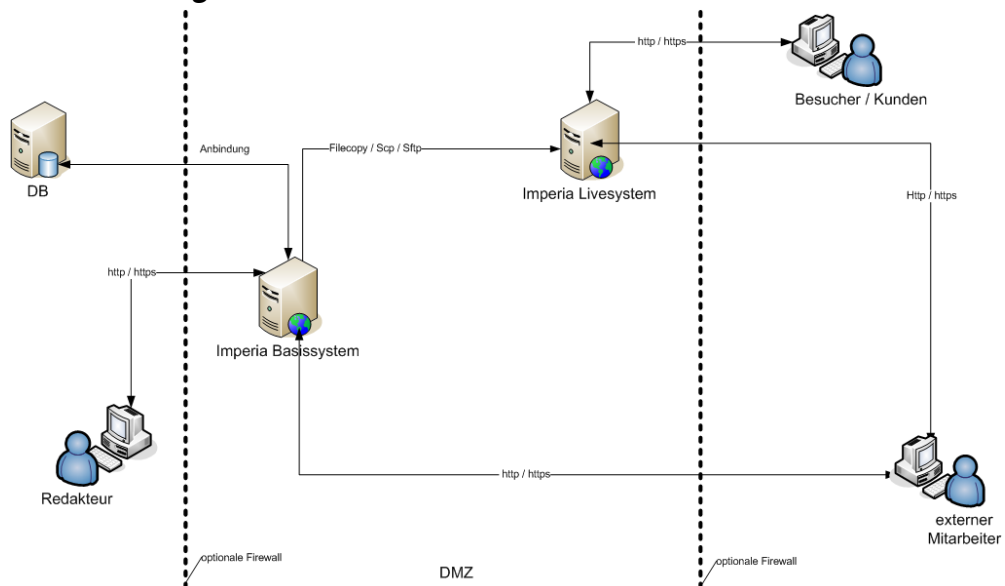
Diese Umgebung ist die übliche Konfiguration für mittlere Systeme.



|                 |                     |
|-----------------|---------------------|
| Prozessoren     | 2 CPUs              |
| Hauptspeicher   | 4GB RAM             |
| Plattenspeicher | 100GB ggf. als RAID |

**Tabelle 4.3. Konfiguration „Standard Filesystem“**

#### 4.5.4 SQL-Datenhaltung



Diese Umgebung ist die Grundkonfiguration für ein System mit einer externen SQL-Datenhaltung.

Der Server für den Betrieb des Webserver wird wie folgt konfiguriert:

|                 |                     |
|-----------------|---------------------|
| Prozessoren     | 2 CPUs              |
| Hauptspeicher   | 4GB RAM             |
| Plattenspeicher | 100GB ggf. als RAID |

**Tabelle 4.4. Konfiguration Webserver „SQL-Datenhaltung“**

Der Server für den Betrieb der Datenbank wird wie folgt konfiguriert:

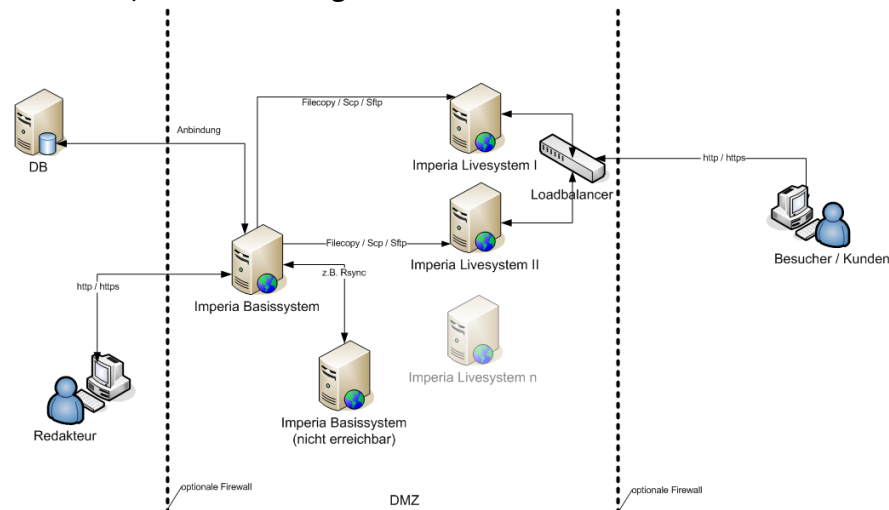
|                 |                     |
|-----------------|---------------------|
| Prozessoren     | 2 CPUs              |
| Hauptspeicher   | 4GB RAM             |
| Plattenspeicher | 100GB ggf. als RAID |

Tabelle 4.5. Konfiguration Datenbankserver „SQL-Datenhaltung“

Diese Umgebung skaliert mit der Hardware der Systeme.

Ausfallsicherheit würde analog zur „Minimalkonfiguration“ durch redundante Standbysysteme realisiert.

#### 4.5.5 Loadbalancer, SQL-Datenhaltung



Diese Umgebung wird für Hochverfügbarkeits-Szenarien eingesetzt. Dabei werden zwei (oder mehr) **imperia** -Server über einen Loadbalancer angesteuert.

Diese Umgebung setzt sich aus den folgenden Komponenten je 2+n **imperia** -Server zusammen:

|                 |                     |
|-----------------|---------------------|
| Prozessoren     | 4 CPUs              |
| Hauptspeicher   | 8GB RAM             |
| Plattenspeicher | 100GB ggf. als RAID |

Tabelle 4.6. Konfiguration Webserver „Loadbalancer, SQL-Datenhaltung“

Der Server für den Betrieb der Datenbank wird wie folgt konfiguriert:

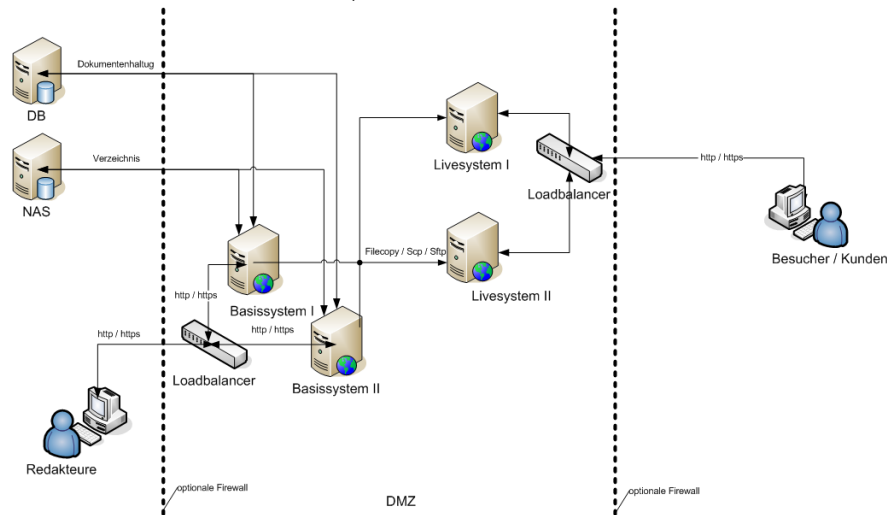
|                 |                     |
|-----------------|---------------------|
| Prozessoren     | 4 CPUs              |
| Hauptspeicher   | 4GB RAM             |
| Plattenspeicher | 100GB ggf. als RAID |
| Loadbalancer    |                     |

Tabelle 4.7. Konfiguration Datenbankserver „Loadbalancer, SQL-Datenhaltung“

Diese Umgebung skaliert mit der Hardware der Einzelsysteme.

Der **imperia** -Dienst „Hermes“ muss auf dem **imperia** -Basissystem laufen und auf jedem der **imperia** -Livesysteme, wenn dort SiteActive-Funktionen benötigt werden.

### 4.5.6 Loadbalancer für Basis- und Livesystem



Soll auch das Basissystem zur Lastverteilung über einen Loadbalancer betrieben werden, so kann dies durch Verwendung eines schnellen NAS erfolgen. Im NAS wird dann die gesamte Struktur des **imperia**-Basissystems abgelegt.



**Hinweis:**

*Wichtig bei der Verwendung mehrerer **imperia**-Basissysteme ist, dass der Hermes nur auf genau einem System laufen darf.*



**Hinweis:**

*Das NAS muss die gleichen Zugriffsgeschwindigkeiten wie ein lokales Filesystem bieten.*

Zur Datenhaltung sollte in diesem Fall unbedingt eine Datenbank verwendet werden. Dies ist sowohl aus Gründen der Datensicherung/Wiederherstellung als auch aus Gründen geringerer Dateizugriffe ins Filesystem angebracht.

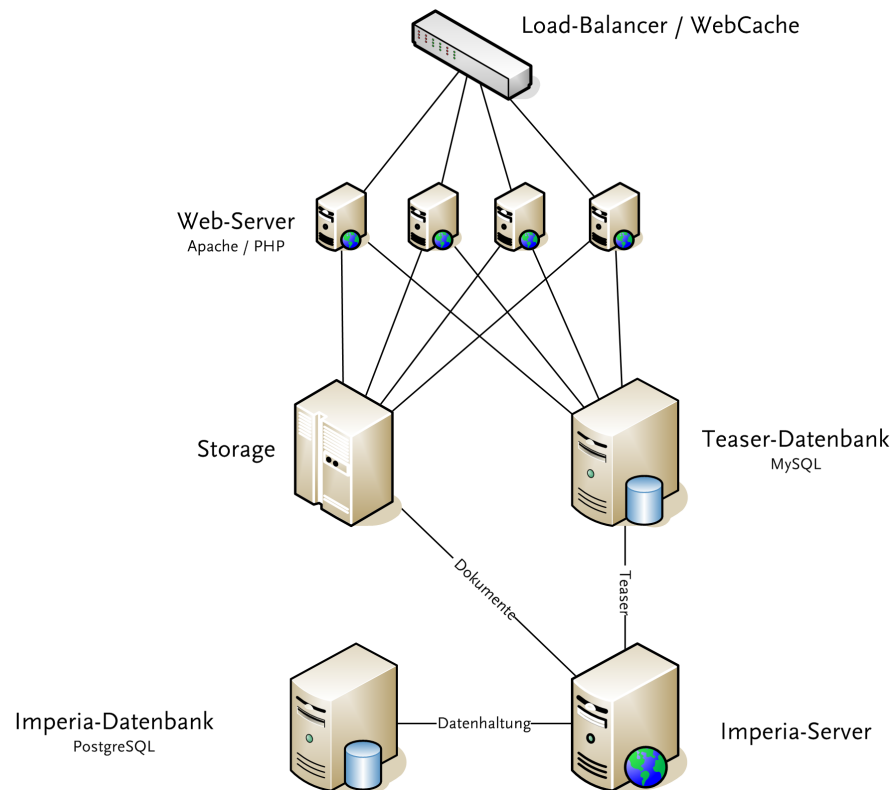
Eine weitere Optimierungsmöglichkeit besteht darin, den Hermes auf einen separaten Server auszulagern. Dies ist bei Systemen mit geplanten lastintensiven Hermes-Operationen (Import, Reparsen, SiteActives) sinnvoll und entlastet das jeweilige Basissystem.

Mit Unterstützung des Professional-Service der Imperia AG können weitere Optimierungen, durch Verlagerung der **imperia**-Module in die lokalen Filesysteme durchgeführt werden.

## 4.6 Systeme mit Teasermanagement

Die folgenden Betriebssysteme basieren auf dem Zusatzmodul „Teasermanagement“.

### 4.6.1 Variante 1



Diese Variante wird von einem Medienunternehmen für die Realisierung der zentralen Auftritte verwendet. (In den Peak-Zeiten zwischen 20.00 Uhr und 22.00 Uhr werden ca. 45.000 Pls/Minute erreicht.)

Mit dem System arbeiten etwa 120 Autoren.

Dabei kommt folgende Hardware zum Einsatz:

- **imperia** -Server (für jeden Auftritt ist eine **imperia** -Instanz installiert):
  - **imperia** -Server: HP 336, 2\*Xeon DualCore 3GHz, 4GB RAM
  - **imperia** -Datenhaltung (PostgreSQL): Sun Netra 440, 4\*Sparc, 16GBRAM, Auslastung: 20%
- Livesystem
  - Web-Server: 8 Server, je 2\*Xeon DualCore 3GHz, 8GB RAM
  - Loadbalancer
  - Webcaches: Bluecoat SG8000
  - Teaser-Datenbank (MySQL): Sun Netra 440, 4\*Sparc, 16GBRAM, Auslastung: 20%-100%

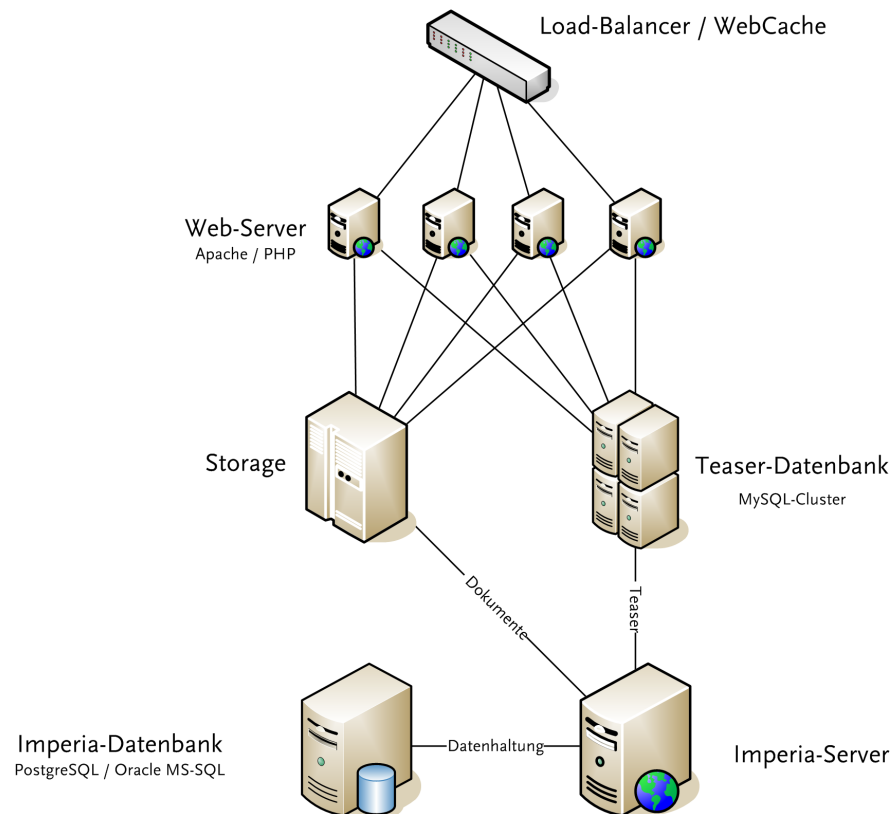
Vorteile:

- Änderungen an Dokumenten/Teasern werden auf dem Livesystem sofort berücksichtigt.

Nachteile:

- Eine Skalierung gestaltet sich schwierig.
- Kaum Puffer für außergewöhnliche Peaks.

## 4.6.2 Variante 2



Abweichend von Variante 1 könnte man anstatt eines großen Servers für die MySQL-Teaser-Datenbank eine Cluster- oder Replikations-Lösung einsetzen.

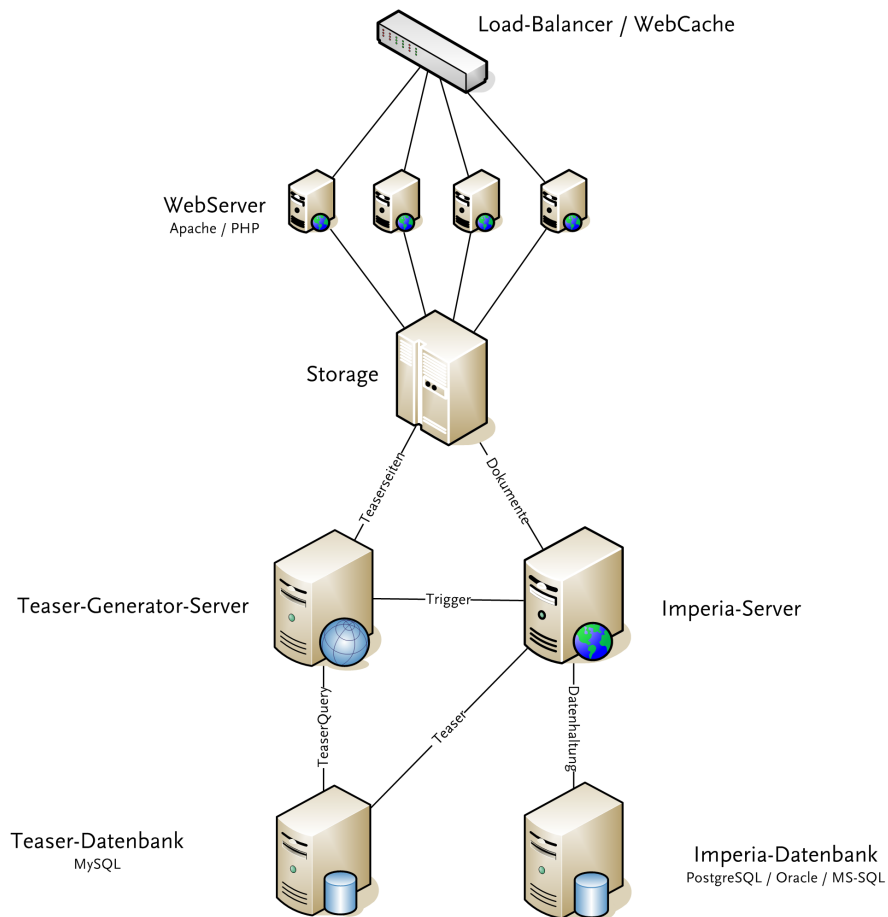
### Vorteile

- Gute Skalierung.

### Nachteile

- Cluster: Einschränkung bei der Datenbankgröße, da die gesamte Datenbank im Hauptspeicher liegen muss.
- Replikation: Erfordert größere Anpassungen der Teasermanagement-Applikation (Schreiben auf Master, Lesen per Loadbalancer).
- Fehleranfälliges Setup.

### 4.6.3 Variante 3



Diese Variante ermöglicht ein zu 100% statisches Livesystem. Das Livesystem greift nicht auf die MySQL-Teaser-Datenbank zu, sondern benutzt nur den Teasermanager-Cache. Diese Teaser-Seiten werden im Hintergrund von einem speziellen Teaser-Generator-Server aktualisiert, indem die Seiten berechnet und neue Cache-Files geschrieben werden. Das **imperia**-System schreibt dazu Aufträge in die Teaser-Datenbank, die der Teaser-Generator-Server abarbeitet. Über Priorisierung der Aufträge könnte so auch eine „Echtzeit-Notfall-Aktualisierung“ realisiert werden.

#### Vorteile

- Teaser-Seiten können bei Bedarf sofort aktualisiert werden.
- Es werden statische, reine PHP-Seiten ohne DB-Zugriff ausgeliefert.
- Gute Skalierung, da nur die Webserver skalieren müssen.

#### Nachteile

- Nicht alle Änderungen sind sofort live sichtbar, sondern in der Regel innerhalb von 1-2 Minuten. Dies hängt davon ab, wie schnell sich im Hintergrund die Seiten berechnen lassen.
- Erfordert Anpassungen der Teasermanagement-Applikation.

## 4.7 Testsysteme

Grundsätzlich ist es für Test- oder Entwicklungssysteme ausreichend, wenn das **imperia**-Entwicklungssystem und das **imperia**-Livesystem auf einem Server installiert werden.

Dabei wird üblicherweise die Datenhaltung verwendet, die auf dem Produktivsystem eingesetzt wird. Dies hat den Vorteil, dass sich das Testsystem in Bezug auf die Datenhaltung genau wie das Produktivsystem verhält.

Bei komplexen Architekturen oder Projekten mit vielen unterschiedlichen Arbeitsgruppen sollte das Testsystem im Aufbau der Produktivumgebung entsprechen. Dies hilft, im Vorfeld eventuelle Schwachstellen aufzudecken.

Neben Testsystemen bei Kunden wird bei **imperia**-Projekten mindestens ein Testsystem bei der Imperia AG aufgesetzt. Dieses System kann dann als Integrationsserver von der Imperia AG und dem Kunden bis zur Abnahme verwendet werden.



#### **Hinweis:**

*Bitte beachten Sie, dass Sie für einen dauerhaften Betrieb eines Testsystems eine eigene **imperia**-Lizenz benötigen.*

## **4.8 Performance**

### **4.8.1 Hardware**

Intel-Architekturen sollten gegenüber Sparc-Architekturen im Zusammenhang mit Webapplikationen, wie einem WCMS, bevorzugt werden, da sie der Anforderung relativ viele komplexe Requests zu beantworten, besser gewachsen sind als die Sparc-Systeme, die effizienter mit kurz laufenden Prozessen umgehen können.

### **4.8.2 Betriebssystem**

### **4.8.3 Filesystem**

Sollte ein Linux-System verwendet werden, so sollte man als Filesystem XFS oder Reiser verwenden.

### **4.8.4 Datenhaltung**

Ist mittelfristig mit mehr als 35.000 Dokumenten (Daumenwert) zu rechnen, so sollte das **imperia** CMS mit einer Datenbank konfiguriert werden. Bei Werten deutlich darunter ist auch das Filesystem in den meisten Fällen gut geeignet. Ausnahmen bestehen wenn z.B. mit sehr vielen Autoren gearbeitet wird oder die Hardware Performanceprobleme hat (z.B. sehr alte Systeme).

### **4.8.5 Browser**

In **imperia 9** werden überwiegend die performanten Browser unterstützt, um die Möglichkeiten aktueller Technologien voll auszunutzen. Alte, langsame Browser, die die aktuellen Techniken nicht unterstützen, können nicht mehr verwendet werden (siehe auch Abschnitt 2.2.2 **Browser** auf Seite 4).

### **4.8.6 Datenstruktur**

Alle bisher aufgeführten Empfehlungen helfen nur dann, wenn man sich auch intensiv Gedanken über die Strukturierung der Daten macht.

Sammelt man all seine privaten Dokumente in einer großen Schublade, so wird das Auffinden des richtigen Dokuments mühsam, bzw. es dauert unter Umständen sehr lange. Bei Dateien im Filesystem und Datensätzen ist dies nicht anders. Man benötigt eine Strukturierung. Ordner! Sowohl für die privaten Dokumente als auch bei Dateien im Filesystem gestaltet sich der Zugriff wesentlich leichter, wenn es Hinweise gibt in welchem Ordner das Dokument abgelegt ist. Bei Datenbanken ist die Strukturierung üblicherweise durch Attribute abgebildet die dazu geeignet sind die Datenmenge beim Zugriff zu verkleinern.

Das Ziel ist, nicht zu viele Dokumente in einem Ordner bzw. einer Rubrik abzulegen. Sollte eine Rubrik deutlich mehr als 1000 Dokument enthalten, sollte man über eine Strukturierung nachdenken.

Sollen z.B. News-Artikel für einen Bereich erstellt werden und es wird mit ca. 25 Artikeln pro Tag gerechnet, so wird die Rubrik nach einem Jahr nahezu 10.000 Dokumente enthalten. Ein Zugriff auf diese Rubrik dauert aber nun relativ lang. Abhilfe: Verzeichnisse z.B. nach Monaten strukturieren. In diesem Fall ergeben sich ca. 750 Dokumente pro Rubrik.

## Kapitel 5. Wartung

Die rein systemseitige Wartung von **imperia** beschränkt sich auf die Erstellung und Überprüfung von Backups und der regelmäßigen Kontrolle der Log-Dateien.

### 5.1 Backup

Die Verzeichnisbäume des Basissystems und die der Livesysteme sollten regelmäßig in einem Backup gesichert werden. Bei der Datenhaltung in einer SQL-Datenbank muss diese im gleichen Rhythmus in ein Backup einbezogen werden.



#### Hinweis:

*Bitte prüfen Sie, ob Sie Ihre Backups auch erfolgreich wieder einspielen können!*

### 5.2 Log-Dateien

Die Log-Dateien des Webservers und die **imperia** Log-Dateien im Verzeichnis `/site/logs` müssen durch die Administration bzw. Applikationsbetreuung auf dem Basissystem und auf den Livesystemen überwacht werden. Erstens, um mögliche Betriebsprobleme zu erkennen und zweitens, um ein mögliches Volllaufen der Festplatte zu verhindern.

Für den Produktiv-Betrieb sollten die Log-Dateien im Verzeichnis `/site/logs` durch ein Log-Rotate abgewechselt werden.

Dabei können die Log-Dateien z.B. jeden Tag in eine neue Datei verschoben und gezippt werden. Nach einigen Tagen können diese Daten dann in ein Archiv verschoben und nach noch längerer Zeit auch wieder gelöscht werden.

# Index

## Symbole

-v, 18  
--help, 18

## A

ALLOW\_TEMPLATE\_FORM, 12  
ALLOW\_TEMPLATE\_FRAME, 12  
ALLOW\_TEMPLATE\_SCRIPT, 12  
Anonyme Variablen, 21  
Apache, 27  
Apache mod\_perl, 23  
Apache-Modul, 23  
Architektur, 35  
Ausführungsschicht, 23  
Autor, 27

## B

Backup, 18, 36  
Basissystem, 22  
BerkeleyDB, 5  
Betriebskonzepte, 1  
Betriebssystem(e)  
  Installation, 2

## C

CGI, 23, 27  
CGI-BIN, 13  
CGI-DIR, 13  
CIFS, 25  
Cookie  
  Gültigkeit, 14  
COOKIE\_TTL, 14

## D

Dateisystem, 3  
Datenbank, 18  
Datenhaltung, 26, 35-36  
DB\_File, 6  
DBD::SQLite, 6  
DEMOMODE, 13  
Demomodus, 13  
Dienstleister, 22  
DOCUMENT-ROOT, 13  
Dokumente, 25  
Dokumente löschen, 27

## E

Entwicklungssystem, 22, 26  
Erstinstallation, 1

## F

Fast-CGI, 24  
Fehlerprotokoll  
  imperia, 14  
Fesptlatte, 25  
File65, 19

Filesystem, 25-26  
Formular(e)  
  im Template, 12  
Frame(s)  
  im Template, 12

## G

Grid  
  Verwendung, 14  
  Voraussetzungen, 14  
Gültigkeit  
  Cookie, 14

## H

Hermes, 18, 21  
HTTP, 14  
HTTP\_HOST, 14  
HTTPS, 14

## I

IIS, 27  
Image-Magick, 27  
ImageMagick, 7  
imperia  
  Fehlerprotokoll, 14  
  Konfiguration, 12  
  verlinkte Objekte, 12  
  Zwischenspeichern, 14  
imperia -Update, 11  
IMPERIA\_ERROR\_LOG, 14  
install.log, 3  
install.pl, 2, 19-20  
  -c(gibin), 2  
  -d(ocroot), 2  
  -f(orce), 3  
  -h(elp), 2  
  -k(eep), 2  
  -l(ive), 2  
  -n(oprompt), 3  
  -p, 3  
  -r(m)(emove), 2  
  -rf / -fr, 3  
  -s(ite), 2  
  -w, 2  
Installation, 1  
  Betriebssystem, 2  
  CGI-BIN-Verzeichnis, 9  
  Document Root, 9  
  ImageMagick, 7  
  IP-Adresse, 9  
  Perl-Link, 9  
  Protokoll, 3  
  UNIX®-Derivaten, 9  
  Update, 11  
  URL, 9  
  User-ID, 9  
  Verzeichnisse, 10-11  
  Voraussetzungen, 3  
  Windows® 2000 / 2003 Server / XP, 10  
  Zielsystem, 11

Installations-Protokoll, 3  
 Installations-Skript, 2  
 Integrationsserver, 35  
 IPlanet, 27

## K

Kommandozeile, 22  
 Konfiguration  
   imperia , 12  
 Konventionen, 1

## L

LATCHING, 14  
 Linux, 27  
 Livesystem, 27  
 Lizenz-Schlüssel, 21  
 Log-Dateien, 36  
 Logrotate, 36

## M

MAM, 18  
 Mandanten trennen, 27  
 MBD, 18  
 Menüs, 21  
 Migration, 1  
 mod\_perl, 23  
 Module, 25  
 MS-SQL, 26  
 MySQL, 26

## N

NAS, 25  
 NFS, 25

## O

OPERATING-SYSTEM, 13  
 Oracle, 26

## P

Partner, 22  
 Perl, 27  
 Perl-Code  
   sichere Umgebung, 14  
 Perl-Link, 9  
 Perl-Modul  
   BerkeleyDB, 5  
   DB\_File, 6  
   DBD::SQLite, 6  
   HTML/FormatText/WithLinks.pm, 6  
   HTML::FormatText::WithLinks, 7  
   HTML::Parser, 6  
   Time::HiRes, 7  
   XML::Dom, 7  
 Perl-Version, 5  
 Port 443, 14  
 Port 80, 14  
 PostgreSQL, 25  
 Produktiv-Betrieb, 36  
 Projekte, 26

Protokoll  
   Installation, 3

## R

Redaktionsserver, 22  
 REG\_KEY, 14  
 REG\_KEY2, 14  
 REG\_NAME, 14

## S

SAFE\_TEMPLATE, 14  
 SAN, 25  
 Schnittstelle, 27  
 SERVER-NAME, 13  
 SERVER-PORT, 14  
 SERVER-SCHEME, 13  
 site\_doctree\_admin.pl, 19-20  
 site\_migrate\_9.pl, 19-21  
 site\_migrate\_9\_file65.pl, 19  
 site\_rebuild\_db.pl, 19, 21  
 Skript(e)  
   im Template, 12  
 Smart Database Update, 15  
   Upgrade und Wartung, 15  
 SMB, 25  
 Solaris, 27  
 Speicherort, 25  
 Staging-Architektur, 22  
 Support, 22  
 system.conf, 12

## T

Template  
   Formular(e), 12  
   Frame(s), 12  
   Skript(e), 12  
 Templates, 21  
 Testsystem, 22, 34  
 Time to live, 14  
 ToDo-Liste, 18

## U

Update, 11, 18  
   installieren, 11  
 UPLOAD\_OBJECTS, 12  
 UTF-8, 18

## V

Verwendung  
   Grid, 14  
 Voraussetzungen  
   Grid, 14  
   Installation, 3  
 Vorbereitung, 18

## W

Wartung, 1, 36  
 Webserver, 7, 18, 21, 23, 27  
 Windows, 27

**X**

XML::Parser, 18

**Z**

Zeichensatz, 18

Zielsystem, 18

    Installation, 11

Zwischenspeichern

    imperia , 14